Fachpraktikum IT-Sicherheit: Dokumentation Dateiserver-Gruppen 5 + 10

Abzugeben am 03.09.2016

Betreuung durch Ralf Naues

Christoph Weißenborn[1], Jörg Ricardo Schumacher[1], Marc Ernst Eddy Woerkom[1], Patrick Häbel[1], Sascha Girrulat[2], Silas Jansen[2], and Waldemar Schmidt[1]

[1]Gruppe 5 (Nord)
[2]Gruppe 10 (Süd)

Inhaltsverzeichnis

Zusammenfassung

Unsere Aufgabe war die Bereitstellung zweier Dateiserver innerhalb der vorgegebenen VPN-Netzwerkumgebung. Dabei waren wir verantwortlich für Organisation, Kommunikation und Dokumentation des Projektes. Unser Projekt hatte kein Budget, keine zentrale Leitung und keine festen Arbeitszeiten. Das agile Verfahren Kanban mit einem Trello-Board hat uns daher sehr geholfen unsere wichtigsten Ziele dennoch zeitgerecht zu erreichen.

Da das Projekt im Rahmen des Fachpraktikums IT-Sicherheit durchgeführt wurde, haben wir ein besonderes Augenmerk auf das Sicherheitskonzept gelegt; der BSI IT-Grundschutz diente uns als Maßstab für die Analyse der Sicherheitsanforderungen.

Bei der technischen Umsetzung hoffen wir mit dem Konfigurationsmanagement-Werkzeug Ansible und einer zentralen Codeverwaltung mittels Github ein solides Fundament für eine mögliche Weiterverwendung der technischen Ergebnisse gelegt zu haben.

Es ist uns in der vorgegebenen Zeit gelungen zwei SambaSMB-Server für die Anwender zur Verfügung zu stellen und 95% aller Sicherheits-Maßnahmen umzusetzen, die wir uns vorgenommen hatten. Alles was wir nicht umsetzen konnten - meist weil uns Zeit, Geld oder Zuarbeit gefehlt hat - ist nicht in Vergessenheit geraten: Im Ausblick haben wir geplante technische Maßnahmen und gewünschte organisatorische Regelungen aufgelistet.

1 Grundlagen

> „Tatsache ist, dass - ohne begleitende Richtlinien oder einen Mechanismus zur Beurteilung - Sicherheit von jedem anders definiert wird und von niemandem verifiziert werden kann. Es gibt keinen Maßstab für eine Übereinstimmung mit einer „Kultur“, und eine „Sicherheitskultur“ wird immer von einer Kultur „erledige die Arbeit“ außer Kraft gesetzt werden.
>
> Wenn es Regeln gibt, schreibe sie auf. Wenn Technologien genutzt werden um die Regeln zu implementieren oder zu überwachen, dann schreibe auch das auf. Wenn Leute die Regeln brechen, lasse Konsequenzen folgen. Wenn die Regeln legitime Arbeiten verhindern, dann ändere sie. So einfach ist das.“
>
> *übersetzt aus „Ten claims that scare security pros“, infoworld.com*

Unsere Gruppen (5 und 10) haben den Auftrag erhalten, jeweils einen Datei-Server einzurichten und in Betrieb zu nehmen. Wir waren selbst dafür verantwortlich das Projekt zu organisieren und die Zusammenrarbeit mit den anderen Gruppen zu regeln, welche für Netz, CA, Mail und Webserver zuständig waren. Das spannendste an diesem Projekt war für uns natürlich die Frage nach der IT Sicherheit. Sie steht im Spannungsfeld zwischen notwendiger ganzheitlicher Betrachtung des Themas und dem beschränkten Auftrag, 2 Dateiserver bereit zu stellen.

Wo wir Risiken gefunden haben, die mit technischen Mitteln alleine nicht zu administrieren sind, haben wir organisatorische Regelungen geschaffen. Wo wir Risiken gefunden haben, die wir alleine nicht absichern können, haben wir stets auf die Kommunikation mit den Beteiligten gesetzt. Und diejenigen Restrisiken, welche wir mit den uns zur Verfügung stehenden Ressourcen an Geld, Zeit und Zuarbeit nicht eliminieren konnten, haben wir zumindest dokumentiert.

Bei unserem Vorgehen haben wir uns in folgender Reihenfolge gefragt:

1. Welche Zusammenarbeit zwischen den Teams Dateiserver Nord und Süd ist notwendig für einen sicheren und effektiven Betrieb?
2. Wie können wir die Kommunikation innerhalb der Dateiserver-Gruppen effizient gestalten?
3. Welche (Teil-)Services müssen wir den Anwendern zur Verfügung stellen?
4. Welche wollen wir darüber hinaus realisieren, wenn noch Zeit bleibt?
5. Woran orientieren wir unser Sicherheitskonzept (d.h. wie wollen wir Sicherheit quantifizieren und beurteilen)?

6. Wie können wir die Kommunikation mit den anderen Teams auf eine zuverlässige Basis stellen? Was benötigen wir von ihnen? Welche Services können wir ihnen anbieten?

2 Projektorganisation

2.1 Formelle Entscheidungsfindung

1. Beschlüsse werden per einfacher Mehrheit gefasst. Dabei erhält jedes anwesende Projektmitglied eine Stimme.

2. Die Abstimmung kann fernmündlich via Mumble[1]/Skype oder schriftlich durch E-Mail/Trello Task erfolgen. Bei einer schriftlichen Abstimmung gilt der Beschluss ab dem Zeitpunkt der nächsten wöchentlichen Mumble-Konferenz.

2.2 Vorgehensmodell

Wir haben uns für die Projektsteuerung mittels **Kanban**[2] entschieden. Mit dem Begriff Kanban bezeichnen wir hier das Projektmanagement-Vefahren, wie es von David Anderson[3] beschrieben wird. Kanban zählt zu den agilen Verfahren. Es weist viele Ähnlichkeiten zu Scrum[4] auf, setzt aber weniger auf formale Anforderungen und Rollen, sondern eher auf allgemeine Prinzipien. Wir haben uns für Kanban entschieden, da es als schlankes (*lean*) Vorgehensmodell nur wenige absolute Anforderungen stellt, die fü ein Projekt ohne festen Arbeitszeiten und mit verteilten Mitarbietern ggf. nicht zielführend sind, uns aber dennoch erlaubt unseren Arbeitsablauf zu definieren und den Arbeitsfluss zu überwachen[5]. Kanban stellt jedoch besondere Anforderungen an die Projektmitarbeiter (PMA): Diese müssen entsprechend geschult, motiviert und kommunikativ sein und dürfen sich nicht an klassische hierarchische Verantwortlichkeiten klammern, wie sie in anderen Vorgehensmodellen häufig vorkommen.

2.2.1 Kanban Praktiken

Im folgenden ist kurz beschrieben, in welcher Form wir die durch David Anderson[6] beschriebenen sechs Praktiken in unserem Projekt konkret umgesetzt haben.

[1] Mum16.
[2] Wik16e.
[3] And11.
[4] Wik16g.
[5] Pat16.
[6] And11.

2.2.1.1 Visualisiere den Fluss der Arbeit: Die Wertschöpfungskette mit ihren verschiedenen Prozessschritten (zum Beispiel Anforderungsdefinition, Programmierung, Dokumentation, Test, Inbetriebnahme) muss gut sichtbar visualisiert werden. Dafür wird das Kanban-Board Trello[7] verwendet, auf dem die unterschiedlichen Stationen als Spalten dargestellt werden.

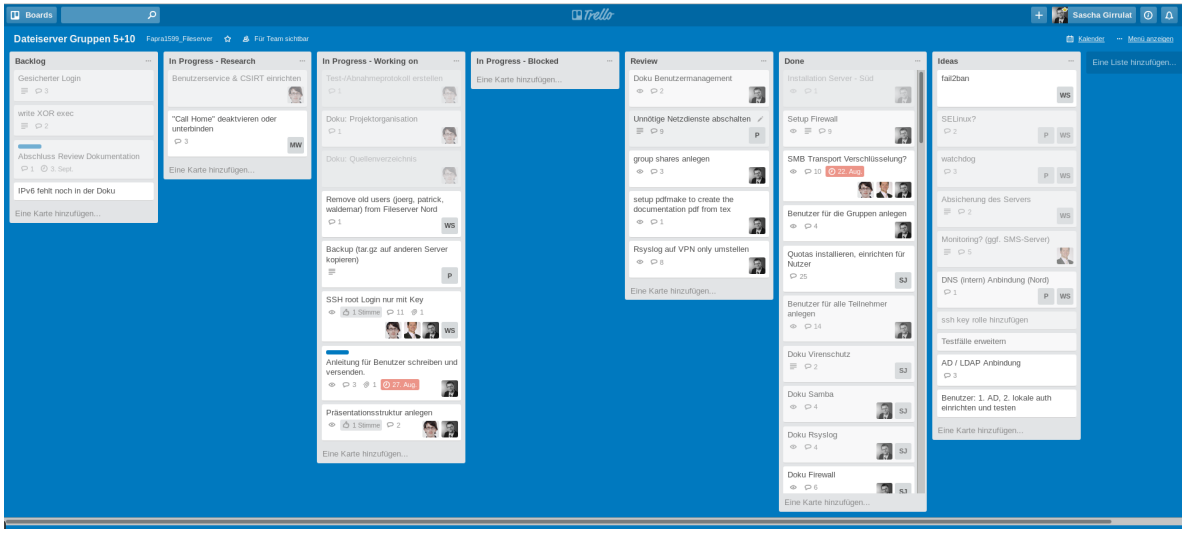

Abbildung 1: Kanban Board Trello für beide Gruppen

Trello unterstützt nicht alle Praktiken, bietet aber ein kostenloses und einfaches Setup für alle Teilnehmer. Um fehlende Funktionen - wie etwa die Begrenzung der angefangenen Arbeit - abzudecken, haben wir an deren Stelle entsprechende organisatorische Regelungen getroffen.

2.2.1.2 Begrenze die Menge angefangener Arbeit: Die Anzahl der Trello Tickets, die gleichzeitig bei einem PMA in Bearbeitung sind, darf bei uns nicht mehr als 2 betragen. Hierdurch entsteht ein Pull-System, bei dem sich jeder PMA seine Arbeit bei der Vorgängerstation abholt, anstatt fertige Arbeit einfach an die nächste Station zu übergeben.

2.2.1.3 Mache die Regeln für den Prozess explizit: Um sicherzustellen, dass alle Beteiligten des Prozesses wissen, unter welchen Annahmen und Gesetzmäßigkeiten man arbeitet, werden möglichst alle Regeln, die es gibt, explizit gemacht. Dazu gehören insbesondere eine Definition des Begriffes „ fertig“, analog der *Definition of Done*[8] (siehe 7.1) in Scrum, sowie eine spezifizierte Abgrenzung der Trello-Spalten untereinander.

[7] Sof16.
[8] Wik16g.

Tabelle 1: Von uns definierte Trello-Kategorien (Spalten)

Bezeichnung	Beschreibung
Backlog	Aufgaben die auf Bearbeitung warten. Hier kann jeder zugreifen und sich eine Aufgabe nehmen. Um zu verhindern, dass Arbeit doppelt gemacht wird, wird das Übernehmen der Aufgabe durch Hinzufügen der eigenen Person zur Aufgabe gekennzeichnet. Da die Zuordnung zu Personen erst in der nächsten Spalte passiert, ist die Taskanzahl hier noch nicht beschränkt.
In Progress - Research	Aufgaben die in Bearbeitung sind und bei denen der Inhalt noch evaluiert werden muss z.B. wenn noch nicht klar ist wie etwas konkret umgesetzt werden muss. Hier sollte die Anzahl, der Aufgaben pro Person, zwei nicht überschreiten.
In Progress - Working on	Aufgaben die konkret in Bearbeitung sind. Hier sollte die Anzahl der Aufgaben pro Person drei nicht überschreiten.
In Progress - Blocked	Die Aufgabe kann nicht weiter bearbeitet werden, etwa weil Zuarbeiten von anderen Personen fehlen oder technische Probleme eine weitere Bearbeitung verhindern. Die Anzahl der Aufgaben sollte pro Person fünf Aufgaben nicht überschreiten.
To Review	Um sicherzustellen, dass alle Aufgaben gemäß DoD mit einem 4-Augen Prinzip bearbeitet wurden, werden bearbeitete Tasks anschließend hierher verschoben. Jeder kann sich hier Aufgaben nehmen und sich den Inhalt entweder vorführen lassen oder selbständig anschauen und ggf. kommentieren oder ergänzen. Sollten Ergänzungen nötig sein muss der Task entsprechend wieder in einen Zustand der Bearbeitung verschoben werden. Die Aufgabenanzahl sollte pro Person drei Aufgaben nicht überschreiten.
Done	Aufgabe ist gemäß DoD abgeschlossen. Taskanzahl ist unbegrenzt.
Ideas	Aufgaben mit niedrigster Priorität, die optional umzusetzen sind. Taskanzahl ist unbegrenzt.

2.2.1.4 Miss und steuere den Fluss: Die Visualisierung und die Begrenzung des Arbeitsflusses sind einfache Mittel, mit denen rasch sichtbar wird, wie schnell die Tickets die verschiedenen Stationen durchlaufen und wo sich Tickets stauen. Die Stellen, vor denen sich Tickets häufen, während an den nachfolgenden Stationen freie Kapazitäten vorhanden sind, werden als Bottlenecks[9] bezeichnet. Durch Analysen des Kanban-Boards

[9] And11.

können immer wieder Maßnahmen ergriffen werden, um einen möglichst gleichmäßigen Fluss (Flow) zu erreichen. Beispielsweise können die Limits für einzelne Stationen verändert werden, es können Puffer eingeführt werden (insbesondere vor Bottlenecks, die durch nur zeitweise Verfügbarkeit von Ressourcen entstehen), die Anzahl der Mitarbeiter an den verschiedenen Stationen kann verändert werden, technische Probleme werden beseitigt usw. Dieser kontinuierliche Verbesserungsprozess (japanisch: Kaizen) ist wesentlicher Bestandteil von Kanban.

Im Rahmen unseres Projektes wurde der Projektfluss bei wöchentlichen Mumble-Meetings besprochen und ggf. Maßnahmen zur Verbesserung eingeleitet. Dabei war die enge Zusammenarbeit der Gruppen 5 und 10 von großer Bedeutung, insbesondere da in Gruppe 10 nur 2 von 5 Kursteilnehmern auch tatsächlich für das Projekt zur Verfügung standen. Weiterhin haben wir unterschiedliche Prioritäten für die Kanban Tasks definiert (siehe 2.4).

Dieser Absatz ist ein Kontrollabschnitt, der sicher stellen soll, dass die vorliegende Projektdokumentation auch tatsächlich von allen Beteiligten gewissenhaft gelesen wurde. Der Verifikationscode lautet hier ”Able Archer 83”.

2.2.1.5 Fördere Leadership auf allen Ebenen in der Organisation: Damit der Prozess der kontinuierlichen Verbesserung funktioniert, ist die Mitwirkung aller Projektbeteiligten erforderlich. Verantwortung sinnvoll übernehmen kann nur, wer auch die ihm übertragene Sache betreffende Entscheidungen treffen und umsetzen darf.

In unserem Projekt sind alle PMA gleichberechtigte Partner. Jeder von uns ist daher berechtigt alle Entscheidungen zu treffen, deren Verantwortung er sich selbst zutraut. Einzige Bedingung ist, dass alle Entscheidungen und Maßnahmen dokumentiert und den anderen PMA zur Überprüfung zur Verfügung gestellt werden.

2.2.1.6 Verwende Modelle, um Chancen für kollaborative Verbesserungen zu erkennen: Da der Fokus des Kurses auf der IT Sicherheit lag, haben wir uns entschlossen, den von uns verwendeten Maßstab für die Sicherheit unserer Lösung (siehe 4.1) auch als Maßstab für den Fortschritt des Projektes zu verwenden.

Durch eine wöchentliche Überprüfung der dazu verwendeten Liste an Maßnahmen haben wir stockende oder noch offene Arbeitspakete identifiziert und bei den regelmäßigen gemeinsamen Besprechungen den dafür notwendigen Arbeitsfluss organisiert.

2.3 Meetings

2.3.1 Status-Meetings

Aufgrund der geografischen Verteilung und der unterschiedlichen Verfügbarkeit der Teammitglieder werden wir die in der agilen Vorgehensweise üblichen, täglichen Statusmeetings in einer etwas geänderten Form abhalten. Alle Teilnehmer der Gruppen Dateiserver Nord und Süd treffen sich Mo,Mi,Fr um 20:00 für ein Statusmeeting von maximal 60 Minuten auf einem der vorhandenen Mumble Server. Hierbei gelten folgende Regeln:

- Das Meetings findet vor dem virtuellen Kanban Board statt
- Jeder sollte seine Sprechzeit zur Status der eigenen Tasks auf 90 Sekunden beschränken
- Jeder sollte innerhalb seiner Sprechzeit folgende Punkte durchgehen:
  - Was habe ich seit dem letzten Treffen gemacht?
  - Was will ich bis zum nächsten Treffen machen?
  - Was blockiert mich bei meiner Arbeit?
- Wenn einer spricht, hört die restliche Gruppe zu
- Tasks sollten wenn möglich innerhalb des Meetings verschoben werden. Wenn die Aufgabenrestriktion der Spalten das nicht zulässt, können Tasks vorher schon auf Review gezogen werden, sollten aber nur innerhalb des Statusmeetings auf Done gezogen werden damit alle einen gemeinsamen Wissensstand haben welche Aufgaben bereits abgeschlossen sind. Am Wochenende werden Statusmeetings nach Absprache gehalten.

2.3.2 Review-Meetings

Einmal in der Woche wird ein Review Meeting durchgeführt, bei dem alle PMA kurz für die Gruppen erläutern, was in der Zwischenzeit geschehen ist. Bei Bedarf können über Screen- oder Teamviewer-Sessions Dinge vorgeführt werden oder Probleme gemeinsam gelöst werden. Die Terminabsprache erfolgt immer innerhalb des Review Meetings für das darauf folgende. Wenn nötig wird ein Protokoll geführt. Informationen hierzu werden im Wiki veröffentlicht.

Wärend des Review Meetings wird auch „Board Grooming“ durchgeführt. Dieser Vorgang ist aus Scrum als *Backlog Grooming*[10] bekannt und beschreibt einen wiederkehrenden

[10] Wik16g.

Prozess zur Überarbeitung und Weiterentwicklung des Backlogs. Wir werden hier ebenfalls einen für Kanban angepassten und vereinfachten Prozess anwenden, der folgende Punkte enthält:

- ordnen und zusammenfassen von Aufgaben auf dem Board
- löschen/beenden von Einträgen, die nicht mehr wichtig sind
- hinzufügen von neuen Einträgen
- detaillieren von Aufgaben
- ggf. nötige Planung von Rollouts
- zuordnen von Reviews
- beenden von abgeschlossenen Tasks

2.4 Priorisierung

Für die Entscheidung, welche Tasks zuerst durchgeführt werden müssen, werden in Kanban häufig die Verzögerungskosten (Cost of Delay) zu Rate gezogen[11]. Da diese bei einem Projekt ohne Budget oder Gewinnabsicht jedoch sehr fiktiv sind, haben wir auf eine formelle Analyse hierzu verzichtet und die Priorisierung bei den wöchentlichen Meetings abgesprochen.

2.4.1 Internes Service Level Agreement (Kanban SLA)

In Kanban sind (optional) verschiedene Service-Arten (Classes of Service) vorgesehen, mit denen die Priorisierung der einzelnen Tasks geregelt werden kann. Wir haben diese leicht verändert auch bei uns realisiert.

[11] Wik16e.

Tabelle 2: Interne SLA Kategorien

Kanban Bezeichnung	Unsere Bezeichnung	Beschreibung
Standard	Standard	Wird automatisch für alle nicht gekennzeichneten Tickets angenommen.
Expedite	Beschleunigt	Aufgaben die mit dieser Priorität versehen werden, gefährden direkt den Erfolg des Projekts und haben Einfluss auf alle Beteiligten. Diese Aufgaben müssen unmittelbar von allen verfügbaren Mitgliedern bearbeitet werden. Direkt im Anschluss muss eine Information der anderen Mitglieder stattfinden und der Sachstand im Task vermerkt werden. Diese Priorität sollte nur in absoluten Ausnahmen vergeben werden. Diese Aufgaben werden mit einer roten Markierung versehen.
Intangible	Optional	Diese Aufgaben werden in der Spalte Ideen gesammelt und mit einer orangen Markierung versehen. Sie sind mit niedriger Priorität zu bearbeiten; solange noch Aufgaben mit höherer Priorität vorhanden sind sollten diese nicht bearbeitet werden. Diese Aufgaben haben keinen Einfluss auf den Grad der Fertigstellung der eigentlichen Projektaufgabe.
Fixed Date	Fester Termin	Wird in der Aufgabe mit einem Fälligkeitsdatum und einer blauen Markierung versehen. Die entsprechenden Tickets sollten so durch das Kanban-System geschleust werden, dass die Funktionalität kurz vor diesem Stichtag produktiv geht.

2.5 Gemeinsames Repository

Um Code und technische Daten auszutauschen nutzen wir ein öffentliches (Git-)Repository[12] auf Github. Um Änderungen durchführen zu können, bekommen alle Mitglieder der beiden Teams (auf Anforderung) entsprechende Berechtigungen.

[12] Git16.

2.5.1 Motivation

Unabhängig von dem expliziten Einsatz von Git bietet ein Versionskontrollsystem uns folgende Vorteile[13]

- **Nachvollziehbarkeit**: Es kann jederzeit nachvollzogen werden, wer wann was geändert hat.
- **Reversibilität**: im Fehlerfall können Änderungen leichter herausgefunden werden und ggf. zurückgenommen werden,
- **Synchronisierung**: gemeinsame Nutzung des Codes durch alle Teilnehmer
- **Kollaboration**: gleichzeitige Arbeit an verschiedenen Features durch Branches
- **Vergleichbarkeit**: durch die Archivierung aller Commits der Teilnehmer sind z.B. problemlos direkte Vergleiche zwischen verschiedenen Versionen möglich.

Wir haben uns für Github entschieden, da es

- kostenlosen Zugang für alle Teilnehmer ermöglicht
- eine integrierte Benutzerverwaltung bietet
- einfache Integration von externen Modulen (fork) vorsieht
- durch Integrationstests via Travis-CI das Testen von Code erleichtert
- ermöglicht die technische Dokumentation durch eine bereits enthaltene Github-zu-Markdown-Schnittstelle einfach vorzubereiten
- Zugriff auch ohne explizite Berechtigungen ermöglicht

2.6 Konfigurationsmanagement-System

Moderne Konfigurationsmanagement-Systeme arbeiten mit einer zentralen Komponente, in welcher der SOLL-Zustand der zu administrierenden Systeme (Zielsysteme) beschrieben wird. Dieses SOLL wird dann durch einen Zugriff auf die Zielsysteme mit dem IST-Zustand abgeglichen und kann auf diesem Wege auch direkt ausgerollt werden.

[13] si616.

2.6.1 Motivation

Ein Konfigurationsmanagement-System bietet unserer Meinung nach - im Vergleich zur lokalen Administration der einzelnen Systeme - folgende Vorteile[14]:

- **Skalierbarkeit**: Neue Server-Systeme können relativ einfach eingerichtet werden; die bestehende Konfiguration kann auf diese Systeme direkt ausgerollt werden
- **Reproduzierbarkeit** und dadurch zeitnahe **Wiederherstellbarkeit**: Kommt es zu nicht nachvollziehbaren Fehlern auf einem Zielsystem, so kann dieses automatisiert wieder in einen definierten Zustand überführt werden. Selbst irreversible Fehler können durch eine komplette Neuinstallation des Systeme deutlich schneller behoben werden: Sobald das Betriebssystem installiert und per ssh erreichbar ist, kann die weitere Software-Installation und Konfiguration vollautomatisch vorgenommen werden. Auch der Austausch von Hardware ist so mit deutlich weniger Aufwand verbunden. In Verbindung mit organisatorischen Regelungen (siehe Betriebshandbuch) ist es uns möglich, mit dem (bei diesem Projekt zwangsläufig beschränkten) Personaleinsatz deutlich kürzere Wiederanlaufzeiten zu realisieren und eine höhere effektive Servicequalität zu gewährleisten.
- **Verifikation** von Code vor der Implementierung: Ansible verfügt über eine „Check"-Option, welche es ermöglicht vor dem Rollout einer Konfiguration zu prüfen, an welchen Stellen genau dadurch eine Änderung auf den produktiven Systemen erfolgt.
- **Systemunabhängigkeit**: Durch die deklarative Beschreibung der gewünschten Konfiguration ist das tatsächlich verwendete Zielsystem bei der Konfiguration transparent; so kann etwa SAMBA mit identischem Code auf zwei verschiedenen Linux-Distributionen ausgerollt werden.
- Vereinigung von technischer **Dokumentation** und Konfiguration: Durch die deskriptive Art der Konfiguration, sowie durch die Möglichkeit Beschreibungen in der Konfiguration einzubauen ist die Konfiguration auch für Personen nachvollziehbar, die nicht mit den technischen Besonderheiten der verwendeten Betriebssysteme und Softwarelösungen vertraut sind. Die Konfiguration dient so gleichzeitig der technischen Dokumentation. In Verbindung mit dem verwendeten Versionskontrollsystem (siehe 3.5) ist dadurch nachvollziehbar, wer wann welche Änderungen genau an den Produktivservern vorgenommen hat.
- Darüber hinaus stellt es bei der Verwaltung von IT-Systeme einen erheblichen **Effizienzgewinn** dar, Konfiguration und Code, zentral zur kollaborativen Bearbeitung zur Verfügung zu haben.

[14] Tow16.

Es standen verschiedene Lösungen, wie z.B. Ansible[15], Puppet[16] und Chef[17] zur Diskussion. Unsere Gruppen haben sich aus folgenden Gründen[18] für den Einsatz von Ansible entschieden:

- Vollumfänglicher Betrieb **ohne Management-Server oder speziellen Agent** auf den Zielsystemen möglich (anders als bei Puppet und Chef)
- Einfache Syntax im yaml (**Markdown**) Format (für Puppet und Chef sind umfangreichere Programmierkenntnisse erforderlich)
- Geringe Anforderungen an das ausführende System - Es ist **lediglich python und ein ssh Client nötig** (Puppet und Chef benötigen z.B. viele Ruby Bibliotheken)
- **kompatibel** mit allen verbreiteten Unix-Systemen (Linux, OpenBSD , Solaris,...)
- „Push-Prinzip“ durch **Ausführung über SSH**; dadurch sind Releases leicht kontrollierbar und die Systeme einfach zugänglich und direkt integrierbar (Puppet und Chef Arbeiten nach einem „Pull-Prinzip“)

Eine spätere Aufnahme von allen Systemen aus den Bereichen Nord und Süd ist nicht nur denkbar, sondern wird von uns auch ausdrücklich empfohlen. Das manuelle Installieren und Konfigurieren von Software in und auf produktiven Systemen ist ein in der Softwareentwicklung häufig anzutreffender schlechter Lösungsansatz für ein bestimmtes Problem (Antipattern[19]).

2.6.2 Verschlüsselung kritischer Codeblöcke

Ansible verfügt über die Möglichkeit schützenswerte Daten über PyCrypto zu verschlüsseln. Dieses Feature nennt sich *Vault* und ermöglicht es alle Variablen in der Ansible üblichen Syntax in AES-verschlüsselten Dateien zu speichern. Diese werden dann zur Laufzeit der Konfiguration in den lokalen Kontext integriert. Hierzu muss ein „Vault-Password“ übergeben werden.

Alle schützenswerten Daten (z.b. die Zertifikate, Initiale Benutzerpasswörter und Schlüssel) werden im Repository durch ein Vault verschlüsselt und mit einem entsprechend komplexen Passwort versehen. Dieses ist allen Teilnehmern bekannt. Bei Bedarf kann das Password PGP-verschlüsselt zur Verfügung gestellt werden und die verschlüsselten Dateien können mit einem neuen Passwort aktualisiert werden.

[15] DeH16.
[16] Lab16.
[17] Che16.
[18] Dre16.
[19] HF10, S. 5-9.

Code 1: Auszug aus ansible/group_vars/file_server/public

```
firstname: sascha
groups: sudo,file-sued,sshlogin,fapra1599,users
lastname: girrulat
name: sgirrulat
passwords:
  crypt: '{{ _vault_user_crypt_password["sgirrulat"] }}'
  plain: '{{ _vault_user_plain_password["sgirrulat"] }}'
```

Code 2: Modifizierter Auszug aus ansible/group_vars/file_server/vault

```
__vault_user_crypt_password:
  sgirrulat:        'xxxxxxx'

__vault_user_plain_password:
  sgirrulat:        'xxxxxxx'
```

Da sbmpasswd in Zusammenarbeit mit lokalen Benutzer nur mit Passwörtern in Klartext umgehen kann und Ansible nur mit Passwörter im Crypt Format, mussten für alle Nutzerkonten die entsprechenden Passwörter in beiden Formaten erzeugt und gespeichert werden.

Tabelle 3: Übersicht der technischen Werkzeuge

Werkzeug	Verwendung für
Trello	• Arbeitspaketierung • Visualisierung des Arbeitsflusses (Kanban) • Arbeitsflusssteuerung (Kanban) • Verbesserungsvorschläge (Kanban: Leadership)
DokuWiki	• Sammelstelle für Dokumentation • Meeting-Protokolle • Austausch komplexer Informationen mit anderen Gruppen
Github Repository	• Zentrale Ablage aller Konfigurationen • Konfliktbehebung bei paralleler Code-Bearbeitung
Travis CI	• Integrationstests
Mumble (VoIP)	• Telekonferenzsystem für Besprechungen • Konfliktbehebung bei paralleler Code-Bearbeitung
Doodle	• Urlaubsplaner
Newsgroup k1599	• multilateraler Austausch mit anderen Gruppen • Anfragen an CIO
Dropbox	• gemeinsame Bearbeitung von Dokumentation und Präsentation • Backup aller Dokumente (enthält nicht den technischen Code auf Github)
TeamViewer	• Videokonferenz für Besprechungen / Präsentationen

2.7 Zusammenarbeit mit anderen Gruppen

FEHLT
- Anfrage von uns an Netz Nord: IPv4, IPv6, DNS, AD
- Antwort von Netz: IPv4-Adressen, kein IPv6, DNS per VPN, AD für Windows only
- Angebot gemeinsames Betriebskonzept blieb unbeantwortet (30.07.16)

2.7.1 Information der Anwender

Alle Kursteilnehmer haben eine mit PGP[20] signierte E-Mail bekommen, in der ihnen zum einen die Daten des persönlichen Accounts mitgeteilt wurden und zum anderen der zugehörige Gruppenbenutzer z.b. ”cert-nord”. Für alle Benutzer wurde ein individuelles Password erzeugt und zugeordnet. Ebenfalls gibt es für jede Gruppe einen eigenen Benutzer, bei dem das Password nur den entsprechenden Gruppen-Mitgliedern bekannt gemacht wurde.
Vorlage:

Sehr geehrte(r) KursteilnehmerIn,

auf den Servern der FileNord und FileSued wurden zwei Benutzer fuer Sie angelegt. Hierbei handelt es sich zum einen um einen personalisierten Benutzer fuer Sie und zum anderen um einen Benutzer fuer ihre Gruppe.

Persoenlicher Benutzer:

name: {{ user_name }} password: {{ user_password }}

Gruppen Benutzer:

name: {{ group_name }} password: {{ group_password }}

Bitte gehen Sie mit diesen Daten sorgsam um, da insbesondere das Passwort des persoenlichen Benutzers nur Ihnen mitgeteilt wird.

Weitere Informationen folgen in separaten Benachrichtigungen.

Viel Erfolg

Gruppe FileNord und Gruppe FileSued

[20] Der öffentliche Schlüssel ist unter `https://pgp.mit.edu/pks/lookup?op=get&search=0x7809DD1F83DCC74A` zu finden.

3 Funktionales Grobkonzept

3.1 Hardware

Aufgrund des nicht vorhandenen Projektbudgets haben wir uns entschieden, zwei Server zu verwenden, welche schon im Vorfeld von unseren Mitgliedern angemietet wurden:
Tabelle 4: Verwendete Server

Service	Bereitgestellt von	Hosting Provider	Betriebssystem
Datei Nord	Jörg Ricardo Schumacher	netcup GmbH	Ubuntu 16.04 LTS
Datei Süd	Sascha Girrulat	Hetzner Online GmbH	Debian GNU/Linux 8.5

Wo sich diese - aus finanziellen Erwägungen getroffene - Entscheidung später auf das Sicherheitskonzept auswirkt, haben wir dies entsprechend dokumentiert.

3.2 Dateiservice

Zur Überlegung standen mehrere Softwarelösungen zur Bereitstellung von Dateidiensten:

- SAMBA (SMB)
- OwnCloud (HTTP/HTTPS/WebDAV/FTP)
- SeaFile (HTTP/HTTPS)
- ProFTPD (FTP)

Wir haben uns schließlich entschlossen SAMBA[21] zu nutzen und den Dateizugriff via SMB/CIFS zu ermöglichen. Für diese Lösung haben wir uns entschlossen, da SAMBA:

- eine erprobte und weit entwickelte Softwarelösung ist
- die Anbindung an ein Active Directory nativ unterstützt
- über die Kommandozeile aus der Ferne administriert werden kann
- Transportverschlüsselung unterstützt (AES-CCM-128)
- für fast alle Unix-Derivate (auf dem Server) verfügbar ist
- SMB/CIFS-Freigaben bietet, die sowohl unter Windows als auch Linux direkt in die Verzeichnisstruktur des Clients eingebunden werden können, so dass aus jeder Anwendung heraus direkt das Speichern und Laden vom Server möglich ist
- auch langfristig keine Lizenzkosten verursacht

[21] Tea16b.

4 Sicherheitskonzept

Es existiert eine Vielzahl von IT-Sicherheitsstandards[22]. Das vorliegende Sicherheitskonzept richtet sich nach den Leitlinien des *BSI IT-Grundschutz*[23].

Kernziel des Projektes ist die Inbetriebnahme zweier Dateiserver. Daher werden im folgenden nur Grundschutzaspekte modelliert, die für den sicheren Betrieb des Dateiservices - unmittelbar oder mittelbar - relevant sind. Die Absicherung anderer Teilaspekte der Gesamtumgebung - wie auch die Absicherung der Gesamtumgebung an sich - bleibt dabei unberücksichtigt. Eine abschließende Zertifizierung nach ISO/IEC 27001 ist nicht Bestandteil des Projektumfangs.

Aus technischer Sicht basiert das Sicherheitskonzept auf der *Philosophie der Verteidigung in der Tiefe (Defense in depth)*[24][25]: Wir verlassen uns nicht auf eine einzelne Sicherheitsmaßnahme (Aus eigener Anschauung kennen wir die Argumentation „Wir brauchen kein gutes Passwort, der Server steht doch hinter der Firewall“ von Administratoren mit sicherheitskritischen Aufgaben), sondern versuchen einem potentiellen Angreifer stets mehrere (voneinander unabhängige) Hürden in den Weg zu stellen. Da aus Ressourcengründen nur jeweils ein Server für Nord und einer für Süd verwendet wird, sind die Möglichkeiten der praktischen Umsetzung hierzu natürlich auf Softwarelösungen beschränkt.

4.1 Security Index

Dem Thema Sicherheit kommt in diesem Projekt eine besondere Bedeutung zu und die Absicherung der Systeme nimmt deutlich mehr Zeit in Anspruch als die Bereitstellung der Funktionalität für den Anwender. Wir haben uns daher entschlossen den Fortschritt des Projektverlaufes daran zu messen, inwieweit wir den Schutzbedarf der Anwendungen erfüllt haben. Um dies an einem einfachen, leicht zu visualisierenden Maßstab messen zu können, haben wir eine eigene Metrik entwickelt: Den Security Index (SI). Dieser ist wie folgt definiert:

Seien

- n die Anzahl der von uns identifizierten BSI IT-Grundschutz Maßnahmen
- m_i die i-te Grundschutz-Maßnahme aus der Tabelle 20 im Anhang

[22] Wik16b.
[23] Sic16.
[24] Wik16c.
[25] KF06.

- $siv(m_i)$ (Security Index Value) das Gewicht [26] der i-ten Maßnahme mit

$$siv(x) = \begin{cases} 0, & \text{wenn Dateiserverguppen nicht verantwortlich für x} \\ 1, & \text{wenn x organisatorische Maßnahme} \\ 2, & \text{wenn x technische Maßnahme von normaler Bedeutung} \\ 4, & \text{wenn x technische Maßnahme von herausragender Bedeutung} \end{cases} \tag{1}$$

- $f(m_i)$ der Grad der Erfüllung der Maßnahme (zwischen 0 und 1), zu entnehmen aus der folgenden Tabelle:

Tabelle 5: Maßnahmenstatus

Maßnahmenstatus	SI Faktor
abgeschlossen	1,0
abgeschlossen mit Restrisiken	0,8
Betriebshandbuch	1,0
externer Dienstl.	1,0
Hardening verboten	1,0
nicht zutreffend	1,0
nur Organisatorisch	0,5
geplant	0,0
nicht durchgeführt	0,0
offen	0,0
Rollout	0,0

dann ist der Sicherheitsindex

$$SI := \frac{\sum_{i=1}^{n} siv(m_i) \times f(m_i)}{\sum_{i=1}^{n} siv(m_i)} \tag{2}$$

Ein SI von $1,0$ bedeutet also, dass alle vom BSI IT Grundschutz vorgeschlagenen Maßnahmen umgesetzt wurden (soweit sie für den Dateiserver-Betrieb relevant sind). Die aktuelle Entwicklung des SI haben wir in unregelmäßigen Abständen - mindestens jedoch

[26] Motivation der Gewichtung: Die tatsächliche Effektivität und Wirksamkeit von technischen Maßnahmen ist leichter zu überprüfen als die von rein organisatorischen Festlegungen („Papier ist geduldig“)

einmal wöchentlich - dokumentiert. Der zum Abgabetermin wirksame SI ist $0,95$.

Der SI kann und soll eine eingehende Betrachtung der aktuellen Sicherheitslage nicht ersetzen. Für eine grobe Abschätzung der IT Sicherheitsvorkehrungen, wie sie etwa im Management erforderlich ist, kann dieser Index jedoch genutzt werden.

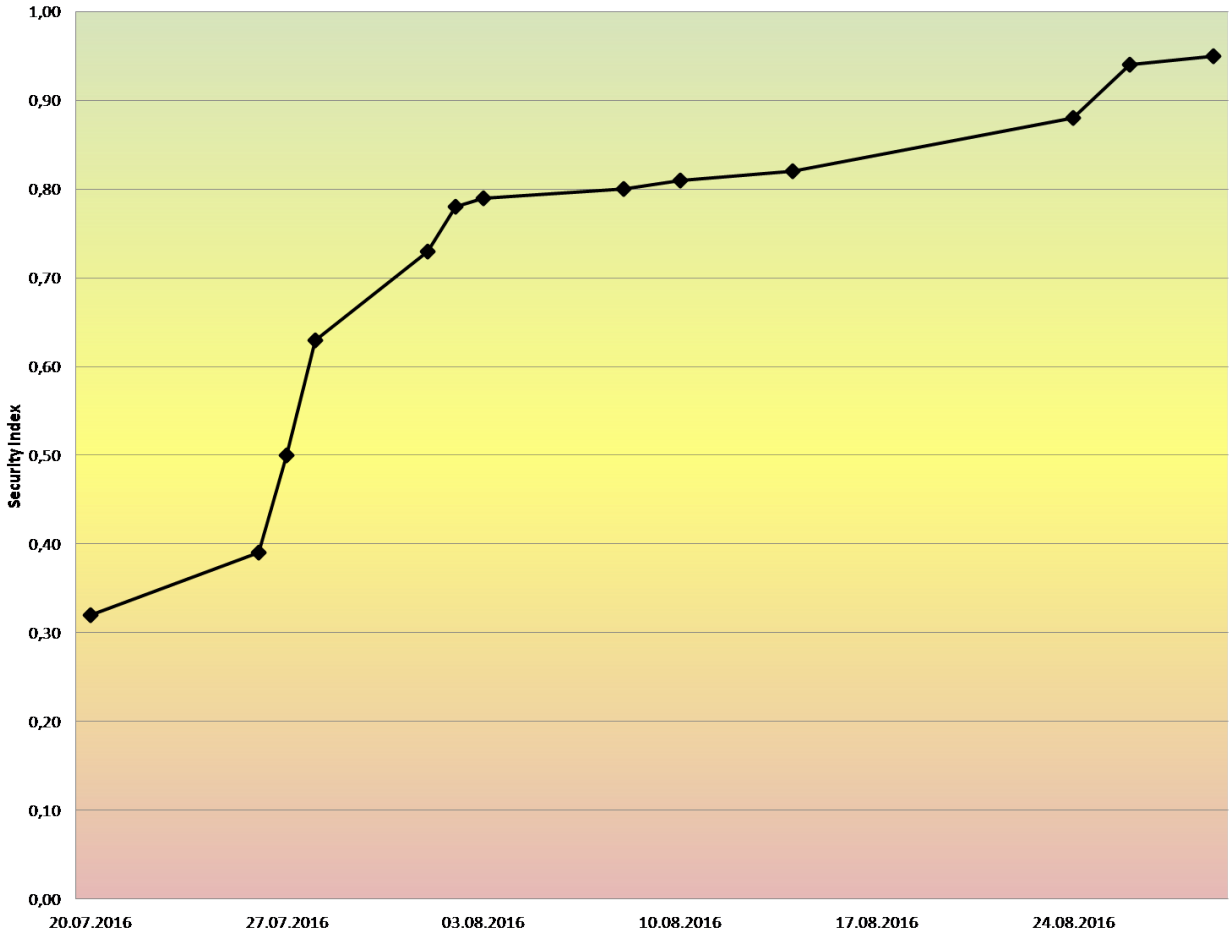

Abbildung 2: Security Index (SI) im Projektverlauf

4.2 Schutzbedarfsfeststellung

Tabelle 6: Definition von Schutzbedarfskategorien

Kategorie	Definition
Normal	Die Schadensauswirkungen sind begrenzt (etwa auf eine einzelne Verkaufsstelle) und überschaubar. Die maximal tragbare Ausfallzeit übersteigt 4 Tage.
Hoch	Die Schadenauswirkungen können beträchtlich sein und z.B. zum Ausfall eines einzelnen Dienstes für alle Verkaufsstellen führen. Die maximal tragbare Ausfallzeit beträgt zwischen 2 und 4 Tagen.
Sehr hoch	Die Schadensauswirkungen können den Bestand der vereinigten Backwerke existentiell bedrohen oder das Betriebsergebnis des Jahres katastrophal Beschädigen. Die maximal tragbare Ausfallzeit liegt unter 2 Tagen

Tabelle 7: Schutzziele und Grundwerte

Grundwert	Definition der Verletzung
Vertraulichkeit	Vertrauliche Informationen werden unberechtigt zur Kenntnis genommen oder weitergegeben
Integrität	Die Korrektheit der Informationen und der Funktionsweise von Systemen ist nicht mehr gegeben
Verfügbarkeit	Autorisierte Benutzer werden am Zugriff auf Informationen und Systeme behindert

4.2.1 Schutzbedarf: Anwendungen

Tabelle 8: Anwendung Dateiserver

Anwendung	Grundwert	Schutzbedarf	Begründung
Personaldaten-verwaltung	Vertraulichkeit	Hoch	Personaldaten sind besonders schutzbedürftige personenbezogene Daten, deren Bekanntwerden die Betroffenen erheblich beeinträchtigen könnte.
Personaldaten-verwaltung	Integrität	Normal	da Fehler rasch erkannt und die Daten nachträglich korrigiert werden können.
Personaldaten-verwaltung	Verfügbarkeit	Gering	Ausfälle von bis zu einer Woche können mittels manueller Verfahren überbückt werden
Finanz-verwaltung	Vertraulichkeit	Hoch	Mit den Finanzdaten ist ein Zugriff auf die Konten der vereinigten Backwerke möglich
Finanz-verwaltung	Integrität	Hoch	Falsche oder fehlende Abrechnungen führen zu unbezahlten oder unbearbeiteten Bestellungen
Finanz-verwaltung	Verfügbarkeit	Normal	Rechnungen müssen nur innerhalb der üblichen Laufzeiten bezahlt werden
Kassensystem	Vertraulichkeit	Gering	Backwerk-Bestellungen sind für unsere Kunden keine vertraulichen Informationen
Kassensystem	Integrität	Normal	Falsche oder fehlende Kassendaten machen die Nachvollziehbarkeit von Ausgaben und Einnahmen unmöglich
Kassensystem	Verfügbarkeit	Normal	Die Synchronisation der Kassenbestände erfolgt nachts und ist für den Tagesbetrieb nicht relevant

4.2.2 Schutzbedarf: IT-Systeme

Tabelle 9: Anwendung Dateiserver

Grundwert	Schutzbedarf	Begründung
Vertraulichkeit	Hoch	Maximumprinzip
Integrität	Hoch	Maximumprinzip
Verfügbarkeit	Normal	Maximumprinzip

4.2.3 Schutzbedarf: Kommunikationsverbindungen

Tabelle 10: Modellierung übergeordneter Aspekte

Anwendung	Grundwert	Schutzbedarf	Begründung
Dateiserver?	Vertraulichkeit	Hoch	Maximumprinzip
LAN-Client	Integrität	Hoch	Maximumprinzip
	Verfügbarkeit	Normal	Maximumprinzip
Dateiserver?	Vertraulichkeit	Hoch	Kommunikation findet hier ausschließlich zur LAN-Anbindung statt
Internet	Integrität	Hoch	
	Verfügbarkeit	Normal	
Dateiserver?	Vertraulichkeit	Hoch	Nutzer-Login-Daten
AD/DNS-Server	Integrität	Hoch	Nutzer-Authentifizierung
	Verfügbarkeit	Normal	

4.2.4 Schutzbedarf: Räume

Entfällt, da hierfür - aus Kostengründen - auf einen bereits festgelegten externen Dienstleister gesetzt wird, bei dem die Konditionen nicht verhandelbar sind (siehe 3.1).

4.3 Bausteine

Tabelle 11: Modellierung übergeordneter Aspekte

Kürzel	Titel	Bemerkungen
B1.3	Notfallvorsorgekonzept	
B1.4	Datensicherungskonzept	
B1.6	Computer-Viren-Schutzkonzept	
B1.7	Kryptokonzept	
B1.8	Behandlung von Sicherheitsvorfällen	Geregelt im Betriebshandbuch

Tabelle 12: Modellierung der Infrastruktur

Kürzel	Titel	Bemerkungen
B2.1	Gebäude	Server stehen in Rechenzentrum der netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd); im Weiteren daher nicht mit betrachtet
B2.2	Elektrotechnische Verkabelung	Server stehen in Rechenzentrum der netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd); im Weiteren daher nicht mit betrachtet
B2.4	Serverraum	Server stehen in Rechenzentrum der netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd); im Weiteren daher nicht mit betrachtet
B2.5	Datenträgerarchiv	Server stehen in Rechenzentrum der netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd); im Weiteren daher nicht mit betrachtet
B2.7	Schutzschränke	Geregelt im Betriebshandbuch Server stehen in Rechenzentrum der netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd); im Weiteren daher nicht mit betrachtet

Tabelle 13: Modellierung der IT-Systeme

Kürzel	Titel	Bemerkungen
B1.3	Notfallvorsorgekonzept	
B1.4	Datensicherungskonzept	
B1.6	Computer-Viren-Schutzkonzept	
B1.7	Kryptokonzept	
B1.8	Behandlung von Sicherheitsvorfällen	Geregelt im Betriebshandbuch

Tabelle 14: Modellierung der Anwendungen

Kürzel	Titel	Bemerkungen
B5.8	Telearbeit	Relevant für die sichere Einbindung der Clients
B5.17	Samba	
B5.18	DNS-Server	Betrieben durch Netzgruppen; im weiteren daher unberücksichtigt
B5.19	Internet-Nutzung	
B5.22	Protokollierung	
B5.23	Cloud Management	Basiert auf IETF Cloud Reference Framework
B5.25	Allgemeine Anwendungen	
B5.26	Serviceorientierte Architektur	

4.4 Maßnahmen

Die Maßnahmen, welche sich aus den modellierten Bausteinen ergeben, sind in der Tabelle 20 „Identifizierte Maßnahmen“ zusammengefasst; dort ist auch der aktuelle Status der Maßnahme dokumentiert. Nähere Informationen (genaue Definition, Fachbegriffe, Beispiele technischer Konfiguration) lassen sich im IT-Grundschutz Handbuch[27] finden.

Alle letztendlich durchgeführten technischen Maßnahmen sind in der technischen Systembeschreibung (siehe 6) aufgeführt. Alle organisatorischen Maßnahmen auf der administrativen Seite sind in den Betriebs- und Serviceprozessen des Betriebshandbuches (im Anhang) dokumentiert; die vom Anwender zu beachtenden Regelungen sind analog in

[27] Sic16.

der „IT Richtlinie für Anwender“ (ebenfalls im Anhang) aufgelistet. Die restlichen Maßnahmen - d.h. jene, die wir analysiert haben, aber im Rahmen des Projektes nicht durchführen konnten - sind im Ausblick (Kapitel 8) aufgelistet.

Tabelle 15: Identifizierte Maßnahmen

Kürzel	Titel	Status	Verantwortung	Bemerkungen
M1.28	Lokale unterbrechungsfreie Stromversorgung	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M2.137	Beschaffung eines geeigneten Datensicherungssystems	nicht durchgeführt	5+10	aus Zeit- und Kostengründen nur teilweise realisiert
M2.138	Strukturierte Datenhaltung	abgeschlossen	5+10	
M2.154	Erstellung eines Sicherheitskonzeptes gegen Schadprogramme	abgeschlossen	5+10	
M2.157	Auswahl eines geeigneten Viren-Schutzprogramms	abgeschlossen	5+10	
M2.158	Meldung von Schadprogramm-Infektionen	Betriebshandbuch	5+10	
M2.159	Aktualisierung der eingesetzten Viren-Schutzprogramme und Signaturen	abgeschlossen	5+10	
M2.160	Regelungen zum Schutz vor Schadprogrammen	abgeschlossen	5+10	
M2.205	Übertragung und Abruf personenbezogener Daten	abgeschlossen	5+10	
M2.22	Hinterlegen des Passwortes	abgeschlossen	5+10	
M2.224	Vorbeugung gegen Schadprogramme	abgeschlossen	5+10	

M2.273	Zeitnahes Einspielen sicherheitsrelevanter Patches und Updates	Hardening verboten	5+10	
M2.31	Dokumentation der zugelassenen Benutzer und Rechteprofile	abgeschlossen	5+10	via Ansible
M2.314	Verwendung von hochverfügbaren Architekturen für Server	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M2.315	Planung des Servereinsatzes	abgeschlossen	5+10	
M2.316	Festlegen einer Sicherheitsrichtlinie für einen allgemeinen Server	Betriebshandbuch	5+10	
M2.317	Beschaffungskriterien für einen Server	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M2.318	Sichere Installation eines IT-Systems	abgeschlossen	5+10	
M2.32	Einrichtung einer eingeschränkten Benutzerumgebung	abgeschlossen	5+10	
M2.33	Aufteilung der Administrationstätigkeiten unter Unix	nicht durchgeführt	5+10	alle in der Gruppe haben volle Admin-Rechte
M2.34	Dokumentation der Veränderungen an einem bestehenden System	Betriebshandbuch	5+10	

M2.35	Informationsbeschaffung über Sicherheitslücken des Systems	Betriebshandbuch	5+10	
M2.351	Planung von Speicherlösungen	abgeschlossen mit Restrisiken	5+10	keine zentralen Mgmt Systeme
M2.354	Einsatz einer hochverfügbaren SAN-Lösung	nicht zutreffend	5+10	Verfügbarkeitsanforderung nicht Sehr hoch
M2.358	Dokumentation der Systemeinstellungen von Speichersystemen	abgeschlossen	5+10	
M2.359	Überwachung und Verwaltung von Speicherlösungen	geplant	5+10	
M2.360	Sicherheits-Audits und Berichtswesen bei Speichersystemen	Betriebshandbuch	5+10	
M2.362	Auswahl einer geeigneten Speicherlösung	abgeschlossen	5+10	
M2.437	Planung des Einsatzes eines Samba-Servers	abgeschlossen	5+10	
M2.438	Sicherer Einsatz externer Programme auf einem Samba-Server	abgeschlossen	5+10	
M2.46	Geeignetes Schlüsselmanagement	nur Organisatorisch	5+10	Server voll, Client nur Org.
M2.525	Erstellung einer Sicherheitsrichtlinie für Speicherlösungen	abgeschlossen	5+10	

M2.526	Planung des Betriebs der Speicherlösung	Betriebshandbuch	5+10	
M2.527	Sicheres Löschen in SAN-Umgebungen	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M2.528	Planung der sicheren Trennung von Mandanten in Speicherlösungen	nicht zutreffend	5+10	Es existiert nur je ein Mandant
M2.529	Modellierung von Speicherlösungen	abgeschlossen	5+10	
M3.23	Einführung in kryptographische Grundbegriffe	abgeschlossen	5+10	alle Anwender sind Admins
M3.54	Schulung der Administratoren des Speichersystems	abgeschlossen	5+10	
M3.68	Schulung der Administratoren eines Samba-Servers	abgeschlossen	5+10	
M3.69	Einführung in die Bedrohung durch Schadprogramme	abgeschlossen	5+10	
M3.92	Grundlegende Begriffe beim Einsatz von Speicherlösungen	abgeschlossen	5+10	
M4.105	Erste Maßnahmen nach einer Unix-Standardinstallation	Hardening verboten	5+10	
M4.106	Aktivieren der Systemprotokollierung	abgeschlossen	5+10	rsyslog
M4.13	Sorgfältige Vergabe von IDs	abgeschlossen	5+10	

M4.14	Obligatorischer Passwortschutz unter Unix	abgeschlossen	5+10		
M4.15	Gesichertes Login	abgeschlossen mit Restrisiken	5+10		ausgenommen Meldung des letzten erfolglosen Login
M4.16	Zugangsbeschränkungen für Benutzer-Kennungen und / oder Terminals	nicht zutreffend	5+10		keine festen Arbeitszeitfenster
M4.17	Sperren und Löschen nicht benötigter Accounts und Terminals	Betriebshandbuch	5+10		
M4.18	Administrative und technische Absicherung des Zugangs zum Monitor- und Single-User-Modus	externer Dienstl.	5+10		
M4.19	Restriktive Attributvergabe bei Unix-Systemdateien und -verzeichnissen	abgeschlossen	5+10		
M4.20	Restriktive Attributvergabe bei Unix-Benutzerdateien und -verzeichnissen	abgeschlossen	5+10		
M4.21	Verhinderung des unautorisierten Erlangens von Administratorrechten	abgeschlossen	5+10		
M4.22	Verhinderung des Vertraulichkeitsverlusts schutzbedürftiger Daten im Unix-System	abgeschlossen	5+10		Login am System beschränkt auf Admins

M4.23	Sicherer Aufruf ausführbarer Dateien	abgeschlossen mit Restrisiken	5+10	PATH nicht überwacht
M4.237	Sichere Grundkonfiguration eines IT-Systems	abgeschlossen	5+10	Integritätsdatenbank = Ansible
M4.238	Einsatz eines lokalen Paketfilters	abgeschlossen	5+10	
M4.239	Sicherer Betrieb eines Servers	Betriebshandbuch	5+10	
M4.24	Sicherstellung einer konsistenten Systemverwaltung	abgeschlossen	5+10	
M4.240	Einrichten einer Testumgebung für einen Server	nicht zutreffend	5+10	Verfügbarkeitsanforderung nicht Sehr hoch
M4.25	Einsatz der Protokollierung im Unix-System	abgeschlossen	5+10	
M4.26	Regelmäßiger Sicherheitscheck des Unix-Systems	nur Organisatorisch	5+10	keine automatisierte Prüfung
M4.274	Sichere Grundkonfiguration von Speichersystemen	abgeschlossen	5+10	
M4.275	Sicherer Betrieb einer Speicherlösung	Betriebshandbuch	5+10	
M4.3	Einsatz von Viren-Schutzprogrammen	abgeschlossen	5+10	
M4.305	Einsatz von Speicherbeschränkungen (Quotas)	Betriebshandbuch	5+10	
M4.326	Sicherstellung der NTFS-Eigenschaften auf einem Samba-Dateiserver	abgeschlossen	5+10	

M4.327	Überprüfung der Integrität und Authentizität der Samba-Pakete und -Quellen	abgeschlossen	5+10	
M4.328	Sichere Grundkonfiguration eines Samba-Servers	abgeschlossen	5+10	
M4.329	Sicherer Einsatz von Kommunikationsprotokollen beim Einsatz eines Samba-Servers	abgeschlossen	5+10	
M4.330	Sichere Installation eines Samba-Servers	abgeschlossen	5+10	
M4.331	Sichere Konfiguration des Betriebssystems für einen Samba-Server	abgeschlossen	5+10	
M4.332	Sichere Konfiguration der Zugriffssteuerung bei einem Samba-Server	abgeschlossen	5+10	
M4.333	Sichere Konfiguration von Winbind unter Samba	nicht zutreffend	5+10	kein AD eingerichtet
M4.334	SMB Message Signing und Samba	abgeschlossen	5+10	da nur als Dateiserver genutzt kann Default bleiben
M4.335	Sicherer Betrieb eines Samba-Servers	abgeschlossen	5+10	
M4.432	Sichere Konfiguration von Serverdiensten	abgeschlossen	5+10	Authentisierung intern unverschlüsselt

M4.433	Einsatz von Datenträgerverschlüsselung	externer Dienstl.	5+10	
M4.435	Selbstverschlüsselnde Festplatten	nicht durchgeführt	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M4.447	Sicherstellung der Integrität der SAN-Fabric	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M4.448	Einsatz von Verschlüsselung für Speicherlösungen	nur Organisatorisch	5+10	at Rest nur in IT Sicherheitsrichtlinie
M4.7	Änderung voreingestellter Passwörter	abgeschlossen	5+10	
M4.80	Sichere Zugriffsmechanismen bei Fernadministration	abgeschlossen	5+10	
M4.84	Nutzung der BIOS-Sicherheitsmechanismen	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M4.85	Geeignetes Schnittstellendesign bei Kryptomodulen	abgeschlossen	5+10	
M4.86	Sichere Rollenteilung und Konfiguration der Kryptomodule	abgeschlossen	5+10	
M4.87	Physikalische Sicherheit von Kryptomodulen	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)

M4.88	Anforderungen an die Betriebssystem-Sicherheit beim Einsatz von Kryptomodulen	abgeschlossen	5+10	
M4.89	Abstrahlsicherheit	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M4.9	Einsatz der Sicherheitsmechanismen von X-Window	abgeschlossen	5+10	X-Window nicht installiert
M4.90	Einsatz von kryptographischen Verfahren auf den verschiedenen Schichten des ISO/OSI-Referenzmodells	abgeschlossen	5+10	
M4.93	Regelmäßige Integritätsprüfung	Betriebshandbuch	5+10	
M4.97	Ein Dienst pro Server	abgeschlossen mit Restrisiken	5+10	Teil der gestellten Anforderungen für Services, Problem für Mumble Nord
M5.10	Restriktive Rechtevergabe	abgeschlossen	5+10	
M5.130	Absicherung des SANs durch Segmentierung	externer Dienstl.	5+10	Server gemietet bei netcup GmbH (Nord), bzw. Hetzner Online GmbH (Süd)
M5.151	Sichere Konfiguration des Samba Web Administration Tools	abgeschlossen	5+10	SWAT nicht installiert
M5.17	Einsatz der Sicherheitsmechanismen von NFS	nicht zutreffend	5+10	

M5.177	Serverseitige Verwendung von SSL/TLS	abgeschlossen	5+10		Teil der gestellten Anforderungen
M5.18	Einsatz der Sicherheitsmechanismen von NIS	nicht zutreffend	5+10		
M5.19	Einsatz der Sicherheitsmechanismen von sendmail	nicht zutreffend	5+10		
M5.20	Einsatz der Sicherheitsmechanismen von rlogin, rsh und rcp	abgeschlossen	5+10		Dienste sollten abgeschaltet sein (Ersatz: ssh)
M5.21	Sicherer Einsatz von telnet, ftp, tftp und rexec	nicht zutreffend	5+10		nicht aktiviert
M5.64	Secure Shell	abgeschlossen	5+10		
M5.72	Deaktivieren nicht benötigter Netzdienste	Hardening verboten	5+10		
M5.9	Protokollierung am Server	abgeschlossen	5+10		
M6.1	Erstellung einer Übersicht über Verfügbarkeitsanforderungen	Betriebshandbuch	5+10		
M6.110	Festlegung des Geltungsbereichs und der Notfallmanagementstrategie	Betriebshandbuch	5+10		
M6.111	Leitlinie zum Notfallmanagement und Übernahme der Gesamtverantwortung durch die Leitungsebene	Betriebshandbuch	5+10		

M6.112	Aufbau einer geeigneten Organisationsstruktur für das Notfallmanagement	Betriebshandbuch	5+10
M6.113	Bereitstellung angemessener Ressourcen für das Notfallmanagement	Betriebshandbuch	5+10
M6.114	Erstellung eines Notfallkonzepts	Betriebshandbuch	5+10
M6.115	Integration der Mitarbeiter in den Notfallmanagement-Prozess	Betriebshandbuch	5+10
M6.116	Integration von Notfallmanagement in organisationsweite Abläufe und Prozesse	Betriebshandbuch	5+10
M6.117	Tests und Notfallübungen	Betriebshandbuch	5+10
M6.118	Überprüfung und Aufrechterhaltung der Notfallmaßnahmen	Betriebshandbuch	5+10
M6.119	Dokumentation im Notfallmanagement-Prozess	Betriebshandbuch	5+10
M6.120	Überprüfung und Steuerung des Notfallmanagement-Systems	Betriebshandbuch	5+10
M6.135	Regelmäßige Sicherung wichtiger Systemkomponenten eines Samba-Servers	Betriebshandbuch	5+10

M6.136	Erstellen eines Notfallplans für den Ausfall eines Samba-Servers	Betriebshandbuch	5+10	
M6.141	Festlegung von Ausweichverfahren bei der Internet-Nutzung	Betriebshandbuch	5+10	
M6.162	Reaktion bei praktischer Schwächung eines Kryptoverfahrens	Betriebshandbuch	5+10	
M6.20	Geeignete Aufbewahrung der Backup-Datenträger	abgeschlossen	5+10	
M6.21	Sicherungskopie der eingesetzten Software	abgeschlossen	5+10	
M6.22	Sporadische Überprüfung auf Wiederherstellbarkeit von Datensicherungen	Betriebshandbuch	5+10	
M6.23	Verhaltensregeln bei Auftreten von Schadprogrammen	Betriebshandbuch	5+10	
M6.24	Erstellen eines Notfall-Bootmediums	abgeschlossen	5+10	Alternatives Vorgehen implementiert: Server neu aufsetzen und config via ansible einspielen
M6.31	Verhaltensregeln nach Verlust der Systemintegrität	Betriebshandbuch	5+10	
M6.32	Regelmäßige Datensicherung	abgeschlossen	5+10	

M6.33	Entwicklung eines Datensicherungskonzepts	abgeschlossen	5+10
M6.34	Erhebung der Einflussfaktoren der Datensicherung	abgeschlossen	5+10
M6.35	Festlegung der Verfahrensweise für die Datensicherung	abgeschlossen	5+10
M6.36	Festlegung des Minimaldatensicherungskonzeptes	abgeschlossen	5+10
M6.37	Dokumentation der Datensicherung	abgeschlossen	5+10
M6.56	Datensicherung bei Einsatz kryptographischer Verfahren	nur Organisatorisch	5+10
M6.96	Notfallvorsorge für einen Server	Betriebshandbuch	5+10
M6.98	Notfallvorsorge und Notfallreaktion für Speicherlösungen	Betriebshandbuch	5+10

5 Datensicherungskonzept

5.1 Zu sichernde Daten

Über das Backup werden alle Benutzerdaten auf den Datei-Freigaben gesichert. Betriebssystem und Software der Server sind **nicht** Bestandteil des Backups[28].
Jeder Nutzer hat ein Quota von 100 MB und das System ist auf maximal 50 Nutzerkonten ausgelegt. Ergo beträgt das maximal zu sichernde *Datenvolumen*

$$dv = 50 \times 100 \text{ MB} = 5000 \text{ MB} < 5 \text{ GB} \tag{3}$$

je Server.

Da wir für unser Netz über keine Erfahrungswerte verfügen gehen wir von einem täglichen *Änderungsvolumen* von

$$cv_{daily} = 0.2 \times dv = 0.2 \times 5000 \text{ MB} = 1000 \text{ MB} \tag{4}$$

aus.

5.2 Datensicherungsplan

Die Datensicherung erfolgt über ein Skript, welches einen tar.gz-Container erstellt. Dieser Container wird über das Netzwerk auf den jeweils anderen Dateiserver kopiert. Er wird mit *GNU Privacy Guard*[29] verschlüsselt. Das entsprechende Passwort wird im Vault abgelegt und ist allen PMA bekannt. Die Ausführung des Skriptes erfolgt über einen Cronjob, welcher jeden Tag um 03:00 CET ausgeführt wird. Es erfolgt aktuell keine Sicherung nach dem Generationenprinzip; das aktuelle Backup überschreibt die Vortagesversion. Verantwortlich für die Datensicherung sind alle PMA. Die Überwachung der Backuplösung wurde im Rahmen des Monitorings geplant, jedoch aus Zeitgründen nicht im Projekt realisiert.

Der aktuelle Textabschnitt ist im Dokument enthalten um absichern zu können, dass der Gesamttext von allen PMA gelesen wurde. Zu diesem Zweck kann bei einer Abfrage der Überprüfungsschlüssel ”Loyal Arrow 09” angegeben werden.

5.3 Restore

Das *Vorgehen* ist im Betriebshandbuch, Standard Change „Restore Dateiserver“ beschrieben. Die gesamte *Rekonstruktionszeit* eines Vollbackups beträgt ca. 4 Stunden

[28] Wie das System im Fehlerfall wiederhergestellt wird ist im Unterabschnitt 5.3 zu finden
[29] Tea16a.

(inkl. Betriebssystem, Software, Daten und Tests). Das ist annehmbar, da es im Rahmen der festgestellten Anforderungen bleibt (siehe 4.2).

Die komplette Neuinstallation des Systems - anstatt nur ein vorher erstelltes System-Abbild wieder zurückzuspielen - ist ungewöhnlich, denn sie beinhaltet immer das Risiko, dass dabei durch die veränderte Software Fehler entstehen, welche erst behoben werden müssen, bevor der Service wieder verfügbar ist. Demgegenüber steht jedoch der Vorteil, dass dann ein System mit aktuellem Softwarestand in Betrieb genommen wird. Da die von uns festgestellten Anforderungen an die Verfügbarkeit des Systems geringer sind als die Anforderungen an die Vertraulichkeit, gehen wir hier Fail-safe[30] vor, d.h. wir stellen einen Service erst dann zur Verfügung, wenn wir auch seine Vertraulichkeit gewährleisten können.

5.4 Minimaldatensicherungskonzept

Siehe Betriebshandbuch, Standard Change „Offline Backup“

[30] Wik16d.

6 Technische Systembeschreibung

Falls nicht explizit getrennt aufgeführt, dann sind die im folgenden aufgeführten Einstellungen auf beiden Dateiservern identisch. Die Softwareinstallation und Konfiguration ist in Ansible hinterlegt. Der vollständige Ansible-Code ist im Anhang zu finden.

6.1 Einführung in die technische Umsetzug mit Ansible

Ansible ist eine auf Python basierende Open-Source Lösung zur Konfiguration und Administration von Linux/Unix Systemen. Es nutzt wahlweise YAML oder JSON zur Zustandsbeschreibung von Systemen. Diese Beschreibung basiert auf einer beliebig kombinierbaren Menge von Rollen, einem Verzeichnis der Zielsysteme und einer Menge von Daten.
Das Verzeichnis der Systeme wird *Inventory* genannt und ist in Gruppen organisiert. Diese werden zur Laufzeit evaluiert und mit Gruppen-, Host- und Rollen bezogenen Daten versehen. Auf diese Daten kann zur Laufzeit innerhalb der Rollen zugegriffen werden. Der Zugriff erfolgt über die *Jinja Templating Engine*. Hierdurch wird eine Trennung von Daten und Code erzeugt die, die Wiederverwendung der erstellten Rollen unterstützt.
Die Rollen enthalten die Konkreten *„Tasks“* die in der Summe die Rollenbeschreibung bilden. Die Ausführung erfolgt sequentiell und ist i.d.R. idempotent, d.h. das Ergebnis ändert sich durch die mehrfache Ausführung hintereinander nicht.

Code 3: Auszug aus ansible/group_vars/file_server/public

```
k1599_file_server_smbd_enabled: yes
```

Code 4: Auszug aus ansible/roles/k1599_file_server/tasks/main.yml

```
name: Ensure package samba is installed
package:
  name: samba

name: Create public dir
file:
  path: "{{ k1599_file_server_public_share_path }}"
  state: directory
  mode: '0777'

name: "Ensure service samba is started and enabled: {{
   k1599_file_server_smbd_enabled }}"
service:
  name: "{{ _smb_srv }}"
  state: started
  runlevel: '2 3 4 5'
  enabled: "{{ k1599_file_server_smbd_enabled }}"
```

Tabelle 17: Von uns definierte Ansible Rollen

Bezeichnung	Beschreibung
k1599_anti_virus	ClamAV Virenscanner
k1599_common	Konfiguration allgemeiner Linux Server
k1599_file_server	Samba
k1599_openvpn_client	OpenVPN Client
k1599_rsyslog_client	RSyslog Client
k1599_rsyslog_server	RSyslog Server
k1599_ssh	OpenSSH
k1599_time_sync	NTP
k1599_users	Benutzerkonten und initiale Passwörter
k1599_quota	Quota (Speicherplatzbeschränkungen)

6.2 OpenVPN

Die CA Zertifikate für Süd und Infos zur Beantragung der Client-Zertifikate sind unter `http://caserv-mueller.westeurope.cloudapp.azure.com/` zu finden, für Nord unter `http://ca-nord.fachpraktikum-1599.de/` . Die Server-Zertifikate und privaten Schlüssel sind verschlüsselt im Ansible Vault abgelegt und werden durch Ansible in den Ordner `/etc/openvpn/certs` kopiert.
Für die Nutzung durch OpenVPN werden diese in der Konfigurationsdatei angeben:

Code 5: Auszug aus /etc/openvpn/client.conf

```
#Zertifikate
#Root-Zertifikat
ca certs/ca-chain.cert.pem
#Eigenes Zertifikat
cert certs/cert.pem
#persoenlicher Schluessel
key certs/privkey.pem
```

Damit der private Schlüssel nur vom Benutzer root zu lesen ist, werden die Rechte auf den privaten Schlüssel durch Ansible entsprechend gesetzt. Die Infos zur Konfiguration von OpenVPN und eine Vorlage für die Konfigurationsdatei ist unter `http://www.belmora.net/fapra1599/` zu finden. Diese wurde mit Variablen versehen, um sie für Nord und Süd nutzbar zu machen, da sich die Konfiguration der OpenVPN-Server unterscheidet, z.B. für:

Code 6: Tunnel-MTU

```
{% if k1599_ovpn_server_mtu != '' %}
#MTU: Fragmentierung, sollte der Einstellung auf dem Server
#entsprechen, daher am besten nicht aendern
tun-mtu {{ k1599_ovpn_server_mtu }}
{% endif %}
```

Code 7: OpenVPN-Server

```
#Server IP und port
remote {{ k1599_ovpn_server }} 1194
```

Code 8: Zertifikats-Betreff

```
{% if k1599_ovpn_server_cn != '' %}
#Authentifizierung des Servers ueber Common Name im Zertifikat
verify-x509-name {{ k1599_ovpn_server_cn }} name
{% endif %}
```

Code 9: Authentifizierungsmethode

```
#Authentifizierung
auth {{ k1599_ovpn_server_auth }}
```

Die Variablen werden durch Ansible bei der Erstellung der Konfigurationsdatei ausgefüllt. Falls der OpenVPN-Client mit `systemctl` gestartet wird, setzt sich der Service Name aus `openvpn@name_der_konfiguration` zusammen (abgelegt in `/etc/openvpn/client.conf`), daher hier `openvpn@client`.

6.3 Samba

Die Konfiguration wird über die Ansible-Rolle *k1599_file_server* durchgeführt. Es stehen für jeden Benutzer eine private Freigabe mit Passwort unter `$benutzername`, sowie eine öffentliche Freigabe `public` für Alle ohne Authentifizierung zur Verfügung. Die zwischen Nord und Süd unterschiedlichen Einstellungen sind in der `smb.conf` mit Variablen versehen, um diese universell nutzbar zu machen. Um Samba auf das OpenVPN-Interface zu beschränken ist die Angabe der IP des Interfaces in der Samba-Konfiguration nötig. tun0 anzugeben funktioniert nicht. Die Variable wird durch Ansible für Nord und Süd entsprechend gefüllt.

Eine Passwortänderung ist für die Benutzer möglich über `smbpasswd -r $servername -U $benutzername`.

Code 10: Ausschnitt aus /etc/samba/smb.conf

```
#### Networking ####

# The specific set of interfaces / networks to bind to
# This can be either the interface name or an IP address/netmask;
# interface names are normally preferred
   interfaces = {{ k1599_file_server_interfaces }}

# Only bind to the named interfaces and/or networks; you must use the
# 'interfaces' option above to use this.
# It is recommended that you enable this feature if your Samba machine is
# not protected by a firewall or is a firewall itself.  However, this
# option cannot handle dynamic or non-broadcast interfaces correctly.
   bind interfaces only = yes
```

Außerdem wird Zugriff ausschließlich aus den bekannten Netzbereichen von Nord und Süd zugelassen und für alle anderen Bereiche verweigert:

Code 11: Ausschnitt aus /etc/samba/smb.conf

```
# allow access only from trusted private networks
   hosts allow = 127.0.0.1 10.8.
   hosts deny = 0.0.0.0/0
```

Zur weiteren Erhöhung der Sicherheit werden einige Protokoll-Optionen gesetzt. NTLM Authentifizierung wird verboten:

Code 12: Ausschnitt aus /etc/samba/smb.conf

```
#### Protocol ####

# allow only NTLMv2
ntlm auth = no
```

Zur Nutzung von Transportverschlüsselung wäre es möglich, das minimale Protokoll auf SMB3 zu beschränken und Verschlüsselung zu erzwingen:

Code 13: Möglicher Eintrag in /etc/samba/smb.conf

```
# set to mandatory for encrypted connection
smb encrypt = auto

# encrypted connection only possible with SMB3
server min protocol = NT1
```

Da dies zum gegenwärtigen Zeitpunkt zu viele Clients ausschließen würde[31] und die Verbindung durch das VPN bereits verschlüsselt ist, haben wir uns dafür entschieden die Transportverschlüsselung aktuell nicht zu erzwingen.

[31] bis einschließlich Windows 7, vgl. [Wik16h]

Unnötige geöffnete Ports werden durch die Deaktivierung von Netbios geschlossen, da Broadcasts über das VPN sowieso nicht weitergeleitet werden:

Code 14: Auszug aus /etc/samba/smb.conf

```
# Disable Netbios
disable netbios = yes
smb ports = 445
```

Damit hört Samba ausschließlich auf Port 445. Der dadurch nicht mehr benötigte Daemon `nmbd` wird erst gar nicht gestartet und durch Ansible deaktiviert.
Die Definition der Freigaben erfolgt für das Home-Directory nach Samba-Standard, diese werden aber zusätzlich beschreibbar gemacht:

Code 15: Auszug aus /etc/samba/smb.conf

```
#======================= Share Definitions =======================

[homes]
   comment = Home Directories
   browseable = no

# By default, the home directories are exported read-only. Change the
# next parameter to 'no' if you want to be able to write to them.
   read only = no

# File creation mask is set to 0700 for security reasons. If you want to
# create files with group=rw permissions, set next parameter to 0775.
   create mask = 0700

# Directory creation mask is set to 0700 for security reasons. If you
    want to
# create dirs. with group=rw permissions, set next parameter to 0775.
   directory mask = 0700

# By default, \\server\username shares can be connected to by anyone
# with access to the samba server.
# The following parameter makes sure that only "username" can connect
# to \\server\username
# This might need tweaking when using external authentication schemes
   valid users = %S
```

Die öffentliche Freigabe soll für jeden Benutzer beschreibbar sein, außerdem soll jeder die dort abgelegten Dateien nutzen dürfen. Da sich der Pfad für die öffentliche Freigabe auf beiden Servern unterscheiden kann, wird dieser in Ansible als Variable definiert.

Code 16: Auszug aus /etc/samba/smb.conf

```
# Add public share
```

```
[public]
path = {{ k1599_file_server_public_share_path }}
read only = no
guest ok = yes
create mask = 0777
directory mask = 0777
```

6.4 Firewall

Als Firewall wird die Linux Kernel Firewall iptables[32] eingesetzt. Zur Umsetzung standen mehrere Implementierungen unter Debian/Ubuntu zur Verfügung, unter anderem *Shorewall*, *apf-firewall* und *ansible-role-firewall*. Das Team hat sich aufgrund der einfachen Integration in den bestehenden Ansible-Code für das externe Playbook `https://github.com/sagiru/ansible-role-firewall` entschieden. Entscheidend war die einfache Integration der Firewall als Service.

Um die benötigten Anforderungen für das Projekt zu erfüllen waren allerdings kleinere Anpassungen nötig. Es fehlten Funktionalitäten wie z.b. das Beschränken von Portfreigaben auf explizite Interface oder die Möglichkeit den Zustand des Services (an/aus) zu Konfigurieren. Diese Funktionalitäten wurden eingebaut und entsprechend in das Upstreamprojekt zurück gegeben.

6.4.1 Konfiguration

Die Iptables Regeln werden über ein Dictionary in Ansible konfiguriert.

Code 17: Auszug aus der Datei ansible/group_vars/file_server_sued/public

```
firewall_allowed_tcp_ports:
  - number: '22'
  - number: '445'
    interfaces:
      - tun0
  - number: 10514
    interfaces:
      - eth0
      - tun0
```

Wenn für einen Port kein Interface angegeben wird, gilt die Konfiguration für alle vorhandenen Interfaces. Der Zugriff wird nur für die nötigen Dienste freigeschaltet und nach Möglichkeit nur über das VPN erreichbar gemacht. Für die Aufbauphase werden die Dienste zusätzlich auf `eth0` freigegeben. Der SSH Zugang bleibt auch für spätere Wartungs-

[32] Net16.

zwecke auf `eth0` verfügbar. Das soll sicherstellen, dass der Server auch im Fehlerfall unabhängig vom VPN gewartet werden kann.

6.4.2 Umsetzung

Die Firewall wurde wie folgt konfiguriert:

- ICMP ist auf allen Interfaces aktiv.
- SSH wird zu Wartungszwecken auf allen Interfaces erlaubt.
- NTP wird auf allen Interfaces zugelassen (durch Openntp eigentlich nicht notwendig)
- Samba wird nur über das VPN erreichbar gemacht.
- Rsyslog ist noch auf allen Interfaces aktiv. Zum Produktivgang ist die Umstellung auf VPN geplant.

Code 18: Ausgabe von iptables -L -v

```
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source
    destination
  408 20400 ACCEPT     all  --  lo     any     anywhere
     anywhere
94393   44M ACCEPT     tcp  --  any    any     anywhere
   anywhere             tcp dpt:ssh
  552 83776 ACCEPT     tcp  --  tun0   any     anywhere
     anywhere             tcp dpt:microsoft-ds
 112K   30M ACCEPT     tcp  --  eth0   any     anywhere
    anywhere             tcp dpt:10514
    0     0 ACCEPT     tcp  --  tun0   any     anywhere
       anywhere             tcp dpt:10514
   84  9632 ACCEPT     icmp --  any    any     anywhere
      anywhere
 9387  713K ACCEPT     udp  --  any    any     anywhere
    anywhere             udp spt:ntp
82952 5716K ACCEPT     all  --  any    any     anywhere
   anywhere             state RELATED,ESTABLISHED
 2570  164K LOG        all  --  any    any     anywhere
    anywhere             limit: avg 15/min burst 5 LOG level debug prefix
     "Dropped by firewall: "
 4593  253K DROP       all  --  any    any     anywhere
    anywhere

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
```

```
 pkts bytes target     prot opt in     out     source
    destination

Chain OUTPUT (policy ACCEPT 284K packets, 35M bytes)
 pkts bytes target     prot opt in     out     source
    destination
 9404  715K ACCEPT     udp  --  any    any     anywhere
    anywhere             udp dpt:ntp
```

Die Regeln werden in ihrer Reihenfolge abgearbeitet. Falls keine zutreffend ist gilt die Default-Regel (hier `ACCEPT`). Um den Zugriff auf nicht freigegebene Dienste in der Input Chain zu verhindern, werden daher durch die letzte Regel alle Pakete verworfen. Die Header-Daten der verworfenen Pakete werden vorher in das Syslog geschrieben.

Code 19: Syslog Eintrag bei verworfenen Paketen

```
fileserver kernel: [312017.164741] Dropped by firewall: IN=eth0 OUT= MAC
   =52:54:a2:01:71:ae:12:54:a2:01:71:ae:08:00 SRC=221.154.167.85 DST
   =172.31.1.100 LEN=40 TOS=0x00 PREC=0x00 TTL=49 ID=61162 PROTO=TCP SPT
   =39130 DPT=23 WINDOW=58288 RES=0x00 SYN URGP=0
```

Außerdem werden eingehende Pakete, die zu ausgehenden Verbindungen gehören zugelassen (`state RELATED,ESTABLISHED`).
Unser System hat keine Routing-Funktionen, die Forward Chain findet deshalb keine Beachtung. In der Output Chain gibt es keine Regel, die Pakete verwirft. Somit sind ausgehende Verbindungen unbeschränkt möglich. Eine spätere Einschränkung ist aber auf die gleiche Weise wie für die Input Chain möglich.

6.5 Virenscanner: ClamAV

Neben dem Paket `ClamAV`[33] als eigentlicher Virenscanner sorgt das Zusatzpaket `freshclam` für die Aktualisierung der Virensignaturen. Das erfolgt automatisch 24x täglich.
Für das gesamte Filesystem ist die Überwachung der Dateien bei Zugriff (on-access scan) aktiviert. Dazu muss ClamAV als `root` ausgeführt werden.

Code 20: Konfiguration in clamav.conf

```
User root
...
ScanOnAccess true
OnAccessMountPath /
```

Bei einem Fund erfolgt ein Eintrag in das Syslog:

[33] **clamav**.

Code 21: Syslog Eintrag bei Virenfund

```
fileserver clamd[569]: ScanOnAccess: /share/public/eicar.com: Eicar-Test-
    Signature(44d88612fea8a8f36de82e1278abb02f:68) FOUND
```

Das Blockieren der Zugriffe ist bei unseren Systemen nicht möglich, da der Kernel die nötigen Erweiterungen nicht mitbringt. Außerdem wird diese Funktion bei der Überwachung des gesamten Dateisystems deaktiviert[34] um die mögliche Blockade des Systems zu verhindern.

6.6 Namensauflösung

Es bestehen verschiedene Möglichkeiten um eine Namensauflösung im Firmennetzwerk zu realisieren. Für die technische Umsetzung wurde von der Gruppe Netzwerke Nord ein zentraler Dienst (DNS) zur Namensauflösung zur Verfügung gestellt; dieser wird auf dem Dateiserver Nord bei der VPN-Einwahl automatisch eingebunden.
Da ein DNS-Server in der Süd-Umgebung nicht realisiert wurde und wir stets eine (so weit wie möglich) einheitliche Konfiguration auf unseren beiden Systemen ausrollen, werden auf beiden Systemen zur Namensauflösung zusätzlich Einträge in der lokalen `/etc/hosts` Datei gepflegt. Die Information welche Einträge vorhanden sein müssen wird durch die jeweilige Netzwerkgruppe zur Verfügung gestellt.
Die Einträge in der Datei `/etc/hosts` werden über die Ansible Rolle *k1599_common* gepflegt.

Code 22: Auszug aus ansible/roles/k1599_common/tasks/main.yml

```
name: Ensure network hosts in /etc/hosts are present
lineinfile:
  dest: '/etc/hosts'
  regexp: '^\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}\s{{ item.name }}\s.*'
  line: "{{ item.ip }} {{item.name}} {{ item.fqdn|default('') }}"
  state: 'present'
with_items:
  - "{{ k1599_common_hosts|default([]) }}"
```

Um die Einträge auf allen verwalteten Systemen synchron zu halten werden die Daten in einem *Dictionary* zur Verfügung gestellt.

Code 23: Auszug aus ansible/group_vars/all

```
k1599_common_hosts:
  - ip: '10.8.3.1'
    name: 'vpn-s'
    fqdn: 'vpn.mueller-backwaren.de'
  - ip: '10.8.3.14'
```

[34] Cis16.

```
    name: 'file-s'
    fqdn: 'fileserver.mueller-backwaren.de'
  - ip: '10.8.3.18'
    name: 'mail-s'
    fqdn: 'mail.mueller-backwaren.de mail.mueller-backwaren.tpweb.de'
  ...
```

Eine Liste der verzeichneten Systeme findet sich in im Anhang.

6.7 Synchronisation der Uhrzeit

Um exakte Zeitstempel für gespeicherte Dateien gewährleisten zu können, benötigt das System die genaue Uhrzeit. Über NTP kann man die lokale Uhr mit anderen Systemen synchronisieren. Das Debian Projekt stellt dazu verschiedene Server in einem Pool zur Verfügung.

Code 24: Debian NTP Server

```
0.debian.pool.ntp.org
1.debian.pool.ntp.org
2.debian.pool.ntp.org
3.debian.pool.ntp.org
```

Das Paket `openntpd` bringt die dafür benötigte Konfigurationsdatei mit und benötigt keine weiteren Einstellungen. Die Wahl ist aus Sicherheitsgründen auf den sehr schlanken openntpd gefallen, da dafür keine Ports von außen erreichbar gemacht werden müssen und dieser auch als reiner Client funktioniert. Die Installation und der Start des Dienstes erfolgt über die Ansible Rolle *k1599_time_sync*.

6.8 Syslog

Um zu unterstützen das keine Betriebs- oder ggf. Sicherheitsrelevanten Informationen verloren gehen stellt jeder der Fileserver einen zentralen Syslogdienst zur Verfügung. Dieser wird über `rsyslog realisiert`. Hierzu wurden die Standardkonfigurationen der Rsyslog-Server durch entsprechende Konfigurationsteile ergänzt. Im ersten Schritt loggen die Dateiserver zusätzlich per tcp auf sich selbst und auf den jeweils anderen Server. Dort wird unter `/var/log/rsyslog-k1599/<serverip>/` eine Dateistruktur erstellt, die dem allgemeinen Standard unter Linux ähnelt.

Code 25: Syslog Dateien

```
root@filesued ~ # ls /var/log/rsyslog-k1599/
138.201.175.250  5.45.103.136

root@filesued ~ # ls /var/log/rsyslog-k1599/138.201.175.250/
```

```
auth.log           auth.log-20160818  daemon.log  kern.log
   messages-20160815  syslog    warn
auth.log-20160815  cron.log           debug       messages
   messages-20160818  user.log
```

Der Rsyslogserver ist so konfiguriert, dass Daten via tcp auf dem Port 10514 angenommen werden. Die Verbindungssicherheit wird über TLS und die durch die Zertifikatsgruppen ausgestellten Schlüssel sichergestellt.

Die Konfiguration wird über die beiden Ansible Rollen *k1599_rsyslog_client* und *k1599_rsyslog_server* vorgenommen. Die Zertifikate der Nord- und Süd-CA sind in `ca-chain.cert.pem` zu finden, damit beide Server ihre Zertifikate gegenseitig als gültig anerkennen.

Code 26: Auszug aus _rsyslog_server/templates/etc/rsyslog.d/30_imtcp_remote_input.conf.j2

```
# Set certificates
global(
  defaultNetstreamDriver="gtls"
  defaultNetstreamDriverCAFile="/etc/rsyslog.d/certs/ca-chain.cert.
     pem"
  defaultNetstreamDriverCertFile="/etc/rsyslog.d/certs/cert.pem"
  defaultNetstreamDriverKeyFile="/etc/rsyslog.d/certs/privkey.pem"
)
```

Außerdem werden die erlautben Gegenstellen durch Überprüfung der Zertifikats-CNs festgelegt.

Code 27: Whitelisting der CN

```
# Enable imtcp listener on port {{ k1599_rsyslog_port }}
module(
  load="imtcp"
  MaxSessions="{{ k1599_rsyslog_maxsessions|default(500) }}"
  StreamDriver.Mode="1" #Nur TLS zulassen
  StreamDriver.AuthMode="x509/name" # Pruefung des CN
  PermittedPeer=["FileNord","fileserver.mueller-backwaren.de"]
)

input(
  type="imtcp"
  port="{{ k1599_rsyslog_port }}"
  ruleset="imtcp_remote_input"
)
...
```

Code 28: Clientseitige Prüfung in /etc/rsyslog.d/10_imtcp_remote_output.conf.j2

```
# Set TLS options
$ActionSendStreamDriverAuthMode x509/certvalid # Pruefe Zertifikat
$ActionSendStreamDriverMode 1 #Nur TLS zulassen
```

Eine Überprüfung der Server-CNs erfolgt nicht, da das aufgrund der fehlenden FQDN beim Zertifikats-CD für Nord nicht funktioniert. Es kann daher nur die Gültigkeit der Zertifikate überprüft werden.

6.9 Benutzerverwaltung

Die Verwaltung der Benutzer erfolgt über die Ansible Rolle *k1599_users*. Hier wird überprüft, ob die Benutzer bereits existieren. Die Benutzer für den Samba Dienst werden über die Rolle *k1599_file_server* verwaltet.

Es wird auf Basis der Datenstruktur `k1599_users_present_users` überprüft, ob die dort angegebenen Benutzer bereits existieren und ggf. angelegt werden müssen. Sollten die Benutzer bereits existieren, so werden sie nicht verändert. So wird sicher gestellt, dass Benutzer ihre Passwörter eigenständig ändern können und diese Änderungen anschließend nicht von Ansible zurückgesetzt werden. Wenn das Aktualisieren der Benutzer explizit gewünscht ist, dann kann auf der Ansible Kommandozeile ein extra Parameter `-e refresh_users=yes` übergeben werden. Hier kann auch durch das überschreiben der Datenstruktur `k1599_users_present_users` (in der Rolle selbst oder durch höher priorisierte Variablen) nur Einfluss auf bestimmte Benutzer genommen werden.

Code 29: Auszug aus ansible/group_vars/file_server/public

```
k1599_users_present_users:
  - firstname: sascha
    groups: sudo,file-sued,sshlogin,fapra1599,users
    lastname: girrulat
    name: sgirrulat
    passwords:
      crypt: '{{ _vault_user_crypt_password["sgirrulat"] }}'
      plain: '{{ _vault_user_plain_password["sgirrulat"] }}'
```

Code 30: Auszug aus der Datei ansible/roles/k1599_users/tasks/main.yml

```
name: Check if user are already created
shell: "getent passwd {{ item.name }} > /dev/null"
register: existing_users
failed_when: false
changed_when: false
with_items:
  - "{{ k1599_users_present_users }}"
```

```
name: Ensure expected users are created
user:
  name: "{{ item.0.name }}"
  password: "{{ item.0.passwords.crypt }}"
  shell: '/bin/bash'
  group: 'users'
  groups: "{{ item.0.groups|default('') }}"
  comment: "{{ item.0.name }}"
  state: present
when: item.1.rc != 0 or refresh_users is defined
with_together:
  - "{{ k1599_users_present_users }}"
  - "{{ existing_users.results }}"
```

6.10 Quota

Mit Filesystem Quotas ist es möglich den für einzelne Benutzer zur Verfügung stehenden Speicherplatz zu beschränken. Dadurch wird sichergestellt, dass jeder Benutzer eine gewisse Menge Platz zur Verfügung hat und kein anderer Benutzer den gesamten Speicherplatz belegen kann. Jedem Benutzer stellen wir 100MB Speicherplatz zur Verfügung, der 7 Tage auf bis zu 150MB überschritten werden kann. Dann werden weitere Schreibzugriffe blockiert.

Damit ein Filesystem auf Quotas überwacht wird, muss dieses mit der Option `usrquota` für Benutzerquota und/oder `grpquota` für Gruppenquota gemountet werden. Wir beschränken uns auf Benutzerquota und die Beschränkung des Speicherplatzes. Die Dateianzahl wird nicht limitiert. Dazu wird `/etc/fstab` wie folgt ergänzt:

```
/dev/sda1 / ext4 rw,relatime,discard,data=ordered,usrquota 0 0
```

Der unter `/aquota.user` abgelegte Quota Index muss initial über den Befehl `quotacheck -vguma` erstellt werden. Sobald das erfolgt ist, werden die Quotas mit `quotaon -av` aktiv gemacht.

Bisher sind noch keine Beschränkungen für die Benutzer festgelegt, über ”repquota /” kann aber bereits eine Übersicht des belegten Speicherplatzes und der Konfiguration ausgegeben werden:

Code 31: Quotasetup

```
*** Report for user quotas on device /dev/sda1
Block grace time: 7days; Inode grace time: 7days
```

```
                        Block limits                File limits
User            used    soft    hard  grace    used  soft  hard  grace
----------------------------------------------------------------------
root      -- 2554096       0       0          32778     0     0
daemon    --      68       0       0              4     0     0
man       --     864       0       0             69     0     0
nobody    --       0       0       0              1     0     0
Debian-exim --      16       0       0              4     0     0
systemd-timesync --       0       0       0              1     0     0
sgirrulat --    3596       0       0             76     0     0
sbruch    --      16       0       0              4     0     0
fhofmann  --      16       0       0              4     0     0
tgrosswendt --      16       0       0              4     0     0
clamav    --  142352       0       0             12     0     0
cweissenborn --      20       0       0              5     0     0
jschumacher --      16       0       0              4     0     0
wschmidt  --      28       0       0              7     0     0
mwoerkom  --      16       0       0              4     0     0
phaebel   --      16       0       0              4     0     0
...
file-sued --      12       0       0              5     0     0
...
```

Die Angabe des belegten Speicherplatzes erfolgt standardmäßig in Kilobyte. Die Konfiguration der qota wird dann über das Hilfsprogramm `quotatool` vorgenommen:

```
quotatool -b -q 102400 -l 153600 -u file-sued /
```

Hiermit werden die Beschränkungen für den Benutzer `file-sued` auf 100MB als *Soft Limit* (Überschreitung maximal 7 Tage) und 150MB als *Hard Limit* gesetzt.
Als Ergebnis von ergibt sich dann für den einzelnen Benutzer:

Code 32: repquota

```
*** Report for user quotas on device /dev/sda1
Block grace time: 7days; Inode grace time: 7days
                        Block limits                File limits
User            used    soft    hard  grace    used  soft  hard  grace
----------------------------------------------------------------------
...
file-sued --      12  102400  153600              5     0     0
...
```

Die Konfiguration der Quotas erfolgt über die Ansible Rolle *k1599_quota*. Damit werden die nötigen Anpassungen am System vorgenommen und die Quotas für alle angelegten Benutzer des Fachpraktikums automatisiert gesetzt.

7 Qualitätssicherung

Das weiter führende Thema Qualitätsmanagement[35] wird bei uns im Rahmen des Betriebskonzeptes (siehe Betriebshandbuch im Anhang) behandelt. Dieser Abschnitt beschreibt daher nur den Teilbereich der Qualitätssicherung.

7.1 Definition of Done (DoD)

Ein Task gilt erst dann als Done (abgeschlossen), wenn er JEDE der folgenden Bedingungen erfüllt:

- Zu jeder abgeschlossenen Aufgabe muss Dokumentation existieren und allen PMA bekannt gemacht werden, oder aber abgelegt sein an einem der folgenden Orte:
  - DokuWiki
  - Dropbox Ordner der Dokumentation
  - im entsprechenden Trello Task
- nach Änderungen im Ansible Code müssen die Integrationstests erfolgreich durchgelaufen sein
- Für neue (nach Task relevante) Features müssen Integrationstests hinzugefügt werden, welche die entsprechende Funktionalität überprüfen z.B. folgende Punkte:
  - Bei einem neuen Dienst sollte überprüft werden, ob dieser seine Funktionalität tatsächlich erfüllt
  - bei einem geöffneten Port sollte überprüft werden, dass dieser nach dem Ansible-Lauf auch offen ist
- die Aufgabe bezieht ihre Relevanz aus dem funktionalen Konzept oder dem Sicherheitskonzept
- fehlgeschlagene Tests dürfen nicht ohne Rücksprache entfernt werden, außer das Feature dazu wird gleichzeitig entfernt
- die Travis-CI Testabdeckung sollte sich nicht verringern
- bei fehlgeschlagenen Tests wird nach dem Schema „You broke it, you fix it!“ vorgegangen
- auf einen fehlgeschlagenen Test/Branch sollten keine neuen Features hinzugefügt werden, hierfür sollte dann ein seperater Branch benutzt werden. Wenn der „master“-Branch dann wieder „grün“ ist, können die andere Features anschließend gemerged werden. Dem liegt die Prämisse „Be always Releasable“ zugrunde

[35] Wik16f.

7.2 Programmier-Regeln

Wir halten uns an die Ansible Dokumentation[36] und die dort empfohlenen „Best Practices“[37]. Eine Ausnahme ist der Aufbau eines Ansible Tasks: Um die Lesbarkeit zu verbessern und den Code zu vereinheitlichen werden diese wie folgt definiert:

Code 33: Formatierung nach Standard

```
- name: be sure ntp is installed
  package: name=ntp state=installed
  tags: ntp
```

Code 34: Unsere verwendete Formatierung

```
- name: Ensure ntp is present
  package:
    name: 'ntp'
    state: 'installed'
  tags: ntp
```

7.3 Automatisierte Testverfahren

7.3.1 Syntaxtests

Um alle relevanten Dateien einer syntaktischen Überprüfung zu unterziehen, wird unter Zuhilfenahme der Sprache *python* entsprechende Funktionalität zur Verfügung gestellt. Hierzu liegt im Repository unter `/ansible` ein Makefile, in dem einige Ziele definiert sind, die es ermöglichen, die syntaktische Korrektheit sicherzustellen. Da die Funktion von Ansible, den Code auf Syntaxfehler zu überprüfen, sich nur auf den gerade ausgeführten Kontext bezieht und nicht auf den vollständigen Code, wird diese Funktion von uns separat zur Verfügung gestellt. Diese Ziele können lokal mittels dem Kommandozeilen Tool `make` ausgeführt werden.

7.3.2 Integrationstests

Da Ansible im Moment noch keine Möglichkeit bietet den Code z.B. Unittests zu unterziehen, nutzen wir die Möglichkeit von Github Integrationstests über die Platform Travis-CI durchzuführen. Hierzu wird im Repository unter
`ansible/tests` eine Testumgebung definiert, welche (so weit wie möglich) einem Fileserver entspricht. Hier werden alle genutzten und zur Erfüllung der Aufgabe benötigten Funktionen getestet. Eingeschränkt testbare Funktionen, wie z.B. der Aufbau der

[36] DeH16.
[37] **ansibleBestPractices**.

```
sascha@discostu:~/git/uni/Wilddiebe10/ansible$ make syntax-checks
for f in $(find host_vars/ group_vars/ -regextype posix-egrep -type f -regex '[^.]+'); do \
  python -c "import yaml; yaml.load(open('$f'))"; \
  if [ $? -ne 0 ]; then \
    echo "Syntax Error in file: $f."; \
    exit 1; \
  fi; \
done
Traceback (most recent call last):
  File "<string>", line 1, in <module>
  File "/usr/lib/python2.7/dist-packages/yaml/__init__.py", line 71, in load
    return loader.get_single_data()
  File "/usr/lib/python2.7/dist-packages/yaml/constructor.py", line 37, in get_single_data
    node = self.get_single_node()
  File "/usr/lib/python2.7/dist-packages/yaml/composer.py", line 36, in get_single_node
    document = self.compose_document()
  File "/usr/lib/python2.7/dist-packages/yaml/composer.py", line 55, in compose_document
    node = self.compose_node(None, None)
  File "/usr/lib/python2.7/dist-packages/yaml/composer.py", line 84, in compose_node
    node = self.compose_mapping_node(anchor)
  File "/usr/lib/python2.7/dist-packages/yaml/composer.py", line 133, in compose_mapping_node
    item_value = self.compose_node(node, item_key)
  File "/usr/lib/python2.7/dist-packages/yaml/composer.py", line 84, in compose_node
    node = self.compose_mapping_node(anchor)
  File "/usr/lib/python2.7/dist-packages/yaml/composer.py", line 127, in compose_mapping_node
    while not self.check_event(MappingEndEvent):
  File "/usr/lib/python2.7/dist-packages/yaml/parser.py", line 98, in check_event
    self.current_event = self.state()
  File "/usr/lib/python2.7/dist-packages/yaml/parser.py", line 550, in parse_flow_mapping_key
    "expected ',' or '}', but got %r" % token.id, token.start_mark)
yaml.parser.ParserError: while parsing a flow mapping
  in "group_vars/all", line 4, column 19
expected ',' or '}', but got '<scalar>'
  in "group_vars/all", line 4, column 33
Syntax Error in file: group_vars/all.
Makefile:33: die Regel für Ziel „check-yaml" scheiterte
make: *** [check-yaml] Fehler 1
```

Abbildung 3: Syntaxprüfung mit unserem Skript

VPN Verbindung, werden im Rahmen der Möglichkeiten getestet. Funktionen, welche eine Netzwerkverbindung zu anderen Systemen benötigen, können auf dem Travis-CI System nicht getestet werden. Die dazugehörigen Testfälle werden im Repository unter `.travis.yml` definiert. Eine Liste der vorhandenen Testfälle findet sich im Anhang. Die Tests werden automatisch beim Hochladen eines neuen Patchsets im Git Repository durchgeführt.

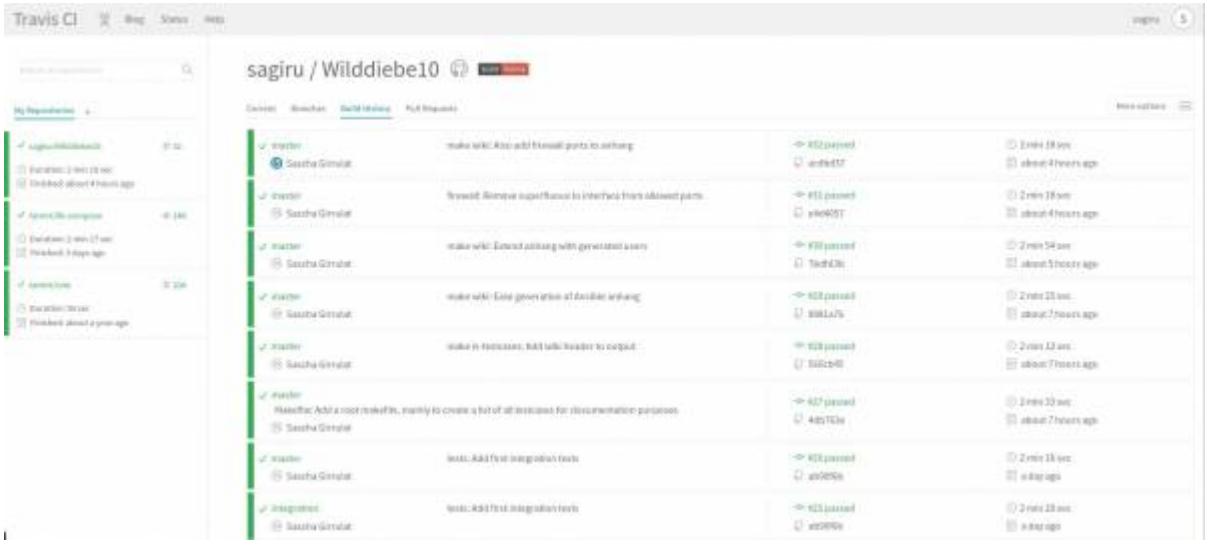

Abbildung 4: Travis CI Oberfläche

8 Ausblick

In diesem Abschnitt werden die Maßnahmen zusammengefasst, die wir als notwendig identifiziert haben, die jedoch im Rahmen des Projekte nicht umgesetzt werden konnten.

8.1 Geplante technische Maßnahmen

- Aufnahme in die Domäne mayerbrot.intern
- Authentisierung in SAMBA gegen das AD, Fallback auf lokale Konten
- Monitoring (Nagios[38]/Check_MK)
  - Überwachung der von uns gewünschten aktiven Prozesse, CPU-/RAM-/HDD-Auslastung
  - Benachrichtigung per Mail per Warnmeldungen und Problemen
  - Visualisierung mit NagVis
  - Aufnahme der Monitoring-Umgebung in Sicherheits- und Betriebskonzepte

8.2 Gewünschte organisatorische Maßnahmen

- IT Sicherheitsbeauftragter
- Vereinheitlichung der Sicherheitskonzepte zwischen den Gruppen
- gemeinsames Betriebskonzept für alle Gruppen (siehe Betriebshandbuch)
- klar festgelegte Risiko Owner für übernommene Risiken
- alle Gruppen umfassende und ausführliche Auslegung des Sicherheitskonzeptes in Vorbereitung auf eine mögliche Zertifizierung (nach welchem Standard dann auch immer[39])

8.3 Blick über den Tellerrand

Auch wenn sich unser Auftrag sehr konkret auf Installation und Betrieb zweier Dateiserver beschränkt, so verstehen wir unsere Aufgabe auch so, dass wir die Gesamtumgebung für die Anwender im Blick behalten sollten. Deshalb haben wir auch Vorschläge gesammelt, die über das hinaus gehen, was von uns verlangt wurde.

[38] Gal16.
[39] Wik16b.

- **VoIP Umgebung** (mögliche Realisierung: Asterisk mit SIP-Clients): Zum Betrieb einer modernen IT-Lösung für Unternehmen gehört auch, sich Gedanken darüber zu machen, wie die Mitarbeiter sicher und effizient miteinander sprechen können. Darüber hinaus gibt es zahlreiche weitere Anforderungen, die nur eine professionelle Lösung abdecken kann, z.B. Callcenter (z.B. für den Benutzerservice), Telekonferenzen, Faxdienste und die Integration mobiler Telefonie.
- Integrierte Sicherheit / **Security by Design**[40]: erst eine aufeinander abgestimmte Client-Server-Architektur, bei der die IT-Abteilung Zugriff auf (und volle Kontrolle über) alle Systeme hat, kann ein wirksames Schutzniveau gewährleisten - und die technische Lösung so gestalten, dass der Anwender das System einfach sicher benutzen kann.

[40] WBM13.

9 Fazit

„Nur Amateure greifen Maschinen an.
Profis zielen auf Menschen.“

Bruce Schneier

9.1 Statement zum Praktikum

Das Thema IT-Sicherheit ist zweifelsohne brandaktuell und übt eine große Faszination selbst auf IT-Laien aus: Vom „Hacken“ in Hollywood-Filmen bis hin zu den zahlreichen Artikeln in Zeitungen und Zeitschriften bei aktuellen Ereignissen, steht die Relevanz des Themas nicht zur Diskussion.

Bei der konkreten Umsetzung des Praktikums haben wir lange über die Anforderungsspezifikation diskutiert: Ist die sehr offene und nur grob umrissene Aufgabenstellung positiv, weil sie die Studierenden zu Selbstverantwortung motiviert und uns ermöglicht, weitgehend selbst Entscheidungen zu treffen? Oder fehlt dem Projekt ein klarer Bezugspunkt: Kein Lastenheft; kein Kunde, an dessen Wünschen man sich orientieren kann und muss; keine Unternehmensleitung, welche die IT-Abteilung auf einen einheitlichen Kurs bringt. Die PMA hatten viele gute Ideen, doch das Fehlen von klaren Hierarchien bedeutet eben auch, dass viele gute Ideen nebeneinander her laufen; daraus entstehen redundante Werkzeuge und Maßnahmen und das vergeudet Ressourcen, die doch in jedem Projekt knapp bemessen sind. Andererseits ging es ja auch gerade darum, aus dem Chaos Ordnung zu erschaffen und eine eigene Organisation aufzubauen, da wo vorher keine war.

Aus technischer Sicht haben wir im Abschnitt 8.3 bereits einige Themen beschrieben, die für zukünftige Praktika interessant sein könnten.

In jedem Fall sind wir dafür dankbar, dass es dieses Praktikum gibt und dass wir in einem Land und zu einer Zeit leben, in der wir uns eingehend mit diesem Thema auseinandersetzen konnten.

9.2 Projekterfolg

Es ist uns in der vorgegebenen Zeit gelungen zwei Datei-Server in Betrieb zu nehmen, die für ihren sicheren Betrieb notwendigen Maßnahmen - seien sie organisatorischer oder technischer Art - umfassend zu analysieren und den größten Teil dieser Maßnahmen anschließend zu implementieren. Nach dem von uns entwickelten Maßstab4.1 haben wir

unser Ziel zu 95% erreicht.

Es bleibt die Frage zu beantworten, was mit den restlichen 5% der Maßnahmen geschehen ist, die sich aus der Sicherheitsanalyse ergeben haben. Hardwarenahe Sicherheitsfunktionen konnten aus Kostengründen nicht realisiert werden; andere technische Verbesserungen - etwa das Monitoring - wurden aus Zeitgründen nicht realisiert. Die von uns vorgesehen organisatorischen Maßnahmen sind vollstädnig implementiert, insoweit wir dafür nicht auf Zuarbeit anderer Gruppen angewiesen waren - von denen es leider oft keine Rückmeldung auf unsere Anfragen gab.
Dennoch: Wäre dies eine produktiv eingesetzte Umgebung, so läge ein großer Teil der Arbeit noch vor uns. Jetzt würde es darum gehen, dem Service den Feinschliff zu verpassen, in der Praxis auftretende Komplikationen zu beheben und die Absicherung iterativ zu verbessern. Da dies jedoch kein Produktivsystem ist, steht in unseren Gedanken der Lerneffekt im Vordergrund, den wir aus diesem Projekt für unser Studium mitnehmen.

Bei einem solch komplexen Projekt bleibt es nicht aus, dass Fehler passieren. Mancher Satz ergibt im Nachgang keinen Sinn, ein Teil der Software-Konfiguration ist lückenhaft oder kompromittiert sogar die Sicherheit. Wir machen uns keine Illusionen: Weder unser System noch unsere Vorstellung davon ist frei von Bugs. Eine Binsenweisheit lautet „Aus Fehlern wird man klug“. Ganz so, wie in der IT-Sicherheit beim Aufdecken von Schwachstellen zunächst alle sich darauf einigen, worum es bei dieser Lücke genau geht[41] und anschließend gemeinsam versuchen sie zu beheben, wünschen auch wir uns, dass wir aus diesem Projekt gemeinsam lernen können, die Zukunft der IT sicher zu gestalten.

[41] Wik16a.

10 Glossar

Tabelle 18: Begriffsdefinitionen im Kontext dieses Projektes

Begriff	Bedeutung im Kontext dieses Projektes
at Rest Verschlüsselung	Verschlüsselung von dauerhaft gespeicherten Daten
CIO	Leiter der IT-Abteilung, aus dem englischen Chief Information Officer. Im Kontext dieses Projektes sind damit die Kursbetreuer gemeint.
Dateiserver Fileserver	Server, der einen festgelegten Bereich eines Dateisystems anderen Benutzern über das Netzwerk zur Verfügung stellt. Der Begriff ist ohne Kontext technisch unspezifisch; er bezieht sich nicht auf ein konkretes Protokoll oder eine Softwarelösung (FTP, SMB, Owncloud, etc.)
Dienst	Ein im Hintergrund laufender Prozess auf einem Computer
Hard Limit	Gibt den maximal nutzbaren Speicherplatz oder die maximale verwendbare Dateianzahl an. Ein Benutzer kann dieses Limit niemals überschreiten. Ein Erreichen dieses Limits ist für den Benutzer gleichzusetzen mit dem Verbrauch des verfügbaren Speicherplatzes
Service	Eine für den Kunden bereitgestellte Servicedienstleistung
Sicherheit sicher abgesichert	Ein IT Service gilt hier als sicher, wenn er die Anforderungen des BSI IT-Grundschutz erfüllt.
Soft Limit	Ist eine Begrenzung, welche kleiner oder gleich dem Hard Limit gesetzt werden muss. Wird das Soft Limit überschritten, so erhält der Benutzer den Zustand Över Quota". Das Soft Limit kann bis zum Wert des Hard Limits überschritten werden
Projektmitarbeiter (PMA)	Studierende in den Gruppen 5 und 10
Anwendungsdaten	Anwendungsspezifische Dateien, ggf. mit Anwenderbezug

Systemdaten	Dateien des Betriebssystems, inklusive Einstellungen ohne Anwenderbezug
Protokolldaten	Daten, die der Überwachung und Nachvollziehbarkeit von Anwendungs- oder Systemereignissen dienen
Anwesenheit anwesend	Als anwesend werden in diesem Projekt alle natürlichen Personen bezeichnet, die - zumindest teilweise - an einer Besprechung teilgenommen haben. Bei fernmündlichen Besprechungen (Mumble, Telekonferenz, etc.) wird als anwesend gezählt, wer am System angemeldet war.
Technische Dokumentation	Die technische Dokumentation umfasst alle Beschreibungen, WIE etwas in technischer Ausführung getan werden muss / getan wurde. Demgegenüber beantwortet das Betriebshandbuch die Frage, WAS getan werden muss.

Literatur

[KF06] David Kuipers und Mark Fabro. *Control systems cyber security: Defense in depth strategies*. United States. Department of Energy, 2006.

[HF10] Jez Humble und David Farley. *Continuous delivery: reliable software releases through build, test, and deployment automation*. Pearson Education, 2010.

[And11] David J Anderson. *Kanban: Evolutionäres Change Management für IT-Organisationen*. dpunkt. verlag, 2011.

[WBM13] Michael Waidner, Michael Backes und Jörn Müller-Quade. *Entwicklung sicherer Software durch Security by Design, Trend-und Strategieberich*. Techn. Ber. Fraunhofer SIT Technical Reports SIT-TR-2013-01, 2013.

[Che16] Chef. *Chef - Embrace DevOps*. 2016. URL: https://www.chef.io/ (besucht am 24. 08. 2016).

[Cis16] Inc. Cisco Systems. *ClamavNet - Configuring On-Access Scanning in ClamAV*. 2016. URL: http://blog.clamav.net/2016/03/configuring-on-access-scanning-in-clamav.html (besucht am 25. 08. 2016).

[DeH16] Michael DeHaan. *Ansible Documentation*. 2016. URL: http://docs.ansible.com/ (besucht am 24. 08. 2016).

[Dre16] Josh Dreyfuss. *Deployment Management Tools: Chef vs. Puppet vs. Ansible vs. SaltStack vs. Fabric*. 2016. URL: http://blog.takipi.com/deployment-management-tools-chef-vs-puppet-vs-ansible-vs-saltstack-vs-fabric/ (besucht am 24. 08. 2016).

[Gal16] Ethan Galstad. *Nagios*. 2016. URL: http://www.nagios.org/ (besucht am 26. 08. 2016).

[Git16] Inc. GitHub. *Github.com, Repository Wilddiebe10*. 2016. URL: https://github.com/sagiru/Wilddiebe10 (besucht am 24. 08. 2016).

[Lab16] Puppet Labs. *Puppet - The shortest path to better software.* 2016. URL: https://www.puppetlabs.com/ (besucht am 24. 08. 2016).

[Mum16] Das Mumble-Team. *Mumble, the open source VoIP solution*. 2016. URL: https://wiki.mumble.info/wiki/Main_Page (besucht am 24. 08. 2016).

[Net16] Netfilter-Projekt-Team. *The netfilter.org "iptables" project*. 2016. URL: https://www.netfilter.org/projects/iptables/index.html (besucht am 25. 08. 2016).

[Pat16] JP Patil. *Get Lean: Why Kanban is the cure to your company's multitasking plague*. 2016. URL: http://venturebeat.com/2013/09/02/why-kanban-is-the-cure-to-your-companys-multitasking-plague/ (besucht am 29. 08. 2016).

[si616] si618. *Stackoverflow.com, Why should I use version control?* 2016. URL: http://stackoverflow.com/questions/1408450/why-should-i-use-version-control (besucht am 24. 08. 2016).

[Sic16] Bundesamt für Sicherheit in der Informationstechnik (BSI). *IT-Grundschutz*. 2016. URL: https://www.bsi.bund.de/DE/Themen/ITGrundschutz/itgrundschutz_node.html/ (besucht am 24. 08. 2016).

[Sof16] Trello Inc. (Fog Creek Software). *Trello.com, Board Dateiserver Gruppen 5-10*. 2016. URL: https://trello.com/b/AQ53ROKH/dateiserver-gruppen-5-10 (besucht am 24. 08. 2016).

[Tea16a] The GNU Privacy Guard Team. *The GNU Privacy Guard*. 2016. URL: https://gnupg.org/ (besucht am 25. 08. 2016).

[Tea16b] The Samba Team. *Samba Wiki*. 2016. URL: https://wiki.samba.org/index.php/Main_Page (besucht am 30. 08. 2016).

[Tow16] Joe Townsend. *Why Is Configuration Management Important?* 2016. URL: https://www.techwell.com/techwell-insights/2013/09/why-configuration-management-important (besucht am 28. 08. 2016).

[Wik16a] Wikipedia. *Common Vulnerabilities and Exposures*. [Online; abgerufen am 25.08.2016]. 2016. URL: https://en.wikipedia.org/wiki/Common_Vulnerabilities_and_Exposures.

[Wik16b] Wikipedia. *Cyber security standards*. [Online; abgerufen am 25.08.2016]. 2016. URL: https://en.wikipedia.org/wiki/Cyber_security_standards.

[Wik16c] Wikipedia. *Defense in depth (computing)*. [Online; abgerufen am 25.08.2016]. 2016. URL: https://en.wikipedia.org/wiki/Defense_in_depth_(computing).

[Wik16d] Wikipedia. *Fail-safe*. [Online; abgerufen am 25.08.2016]. 2016. URL: https://en.wikipedia.org/wiki/Fail-safe.

[Wik16e] Wikipedia. *Kanban (Development)*. [Online; abgerufen am 20.08.2016]. 2016. URL: https://en.wikipedia.org/wiki/Kanban_(development).

[Wik16f] Wikipedia. *Qualitätsmanagement*. [Online; abgerufen am 26.08.2016]. 2016. URL: https://de.wikipedia.org/wiki/Qualit%C3%A4tsmanagement.

[Wik16g] Wikipedia. *Scrum (software development)*. [Online; abgerufen am 24.08.2016]. 2016. URL: https://en.wikipedia.org/wiki/Scrum_(software_development).

[Wik16h] Wikipedia. *Server Message Block - Geschichte*. [Online; abgerufen am 25.08.2016]. 2016. URL: https://de.wikipedia.org/wiki/Server_Message_Block#Geschichte.

A Ansible

A.1 Testfälle

- Run syntax-checks via makefile
- Run the role/playbook via makefile
- Run the role/playbook again to make sure it's idempotent
- Check if rsyslog port 10514 is open
- Check if samba port 445 is open
- Check if ssh port 22 is open
- Check expected users will be created
- Check users will member of expected groups
- Check users will be member of the expected primary group
- Check if user homes got the expected permissions
- Check if groupuser homes got the expected permissions
- Check expected groups will be created
- Check if ssh service is up and running
- Check if samba service is up and running
- Check if samba group share is present
- Check if rsyslog service is up and running
- Check if nmbd service is disabled
- Check if openntpd service is up and running
- Check if quotas are enabled
- Check if quota limits are set as expected
- Check if the doc files are building.

A.2 Ansible Rollen

A.2.1 Eigene Rollen

- k1599_anti_virus
- k1599_common
- k1599_file_server
- k1599_openvpn_client
- k1599_quota
- k1599_rsyslog_client
- k1599_rsyslog_server
- k1599_ssh
- k1599_time_sync
- k1599_users

A.2.2 Externe Rollen

- ansible-role-firewall

A.3 Erzeugte Gruppen

- web-nord
- mail-nord
- netz-nord
- cert-nord
- file-nord
- web-sued
- mail-sued
- netz-sued
- cert-sued
- file-sued

- fapra1599
- sshlogin

A.4 Erzeugte Benutzer

- msiepmann
- shartmann
- tgrosswendt
- netz-nord
- fkamfenkel
- cpempelforth
- mklein
- file-sued
- sgirrulat
- fhofmann
- ahacker
- nnapp
- mwoerkom
- sbruch
- bfischer
- cboettge
- jherold
- fschweisfurth
- web-nord
- jwichert
- mkutter
- phaebel

- cert-nord
- ooffenburger
- mploeger
- dtroeger
- nreusch
- istieglitz
- cweissenborn
- lrichter
- jschumacher
- netz-sued
- swyes
- mail-sued
- msandkuehler
- emueller
- twinkelhorst
- mail-nord
- sganswind
- bwalter
- pholtkamp
- chesseling
- mschrenk
- wpankraz
- file-nord
- ksteinkohl
- tbayer

- mberner
- klauter
- web-sued
- wlindemann
- tschroeder
- msalwitzek
- jaffenzeller
- wschmidt
- aszewc
- mmueller
- sjansen
- cert-sued

A.5 Firewall

A.5.1 Offene Ports - Gruppe Nord

- all: 22
- tun0: 445
- tun0: 10514
- all: 64738

A.5.2 Offene Ports - Gruppe Sued

- all: 22
- tun0: 445
- tun0: 10514

A.6 Einträge /etc/hosts

- 10.8.3.1 : vpn-s vpn.mueller-backwaren.de
- 10.8.3.14 : file-s fileserver.mueller-backwaren.de
- 10.8.3.18 : mail-s mail.mueller-backwaren.de mail.mueller-backwaren.tpweb.de
- 10.8.3.22 : web-s web.mueller-backwaren.de
- 10.8.3.26 : ca-s ca.mueller-backwaren.de
- 10.8.1.1 : vpn-n vpn.mayer-brot.de vpnnord.mayerbrot.intern
- 10.8.1.20 : web-n web.mayer-brot.de webnord.mayerbrot.intern
- 10.8.1.30 : file-n file.mayer-brot.de filenord.mayerbrot.intern
- 10.8.1.40 : ca-n ca.mayer-brot.de canord.mayerbrot.intern
- 10.8.1.50 : mail-n mail.mayer-brot.de mailnord.mayerbrot.intern
- 10.8.2.1 : gateway-n
- 10.8.2.2 : gateway-s

B Testprotokoll

Tabelle 20: Identifizierte Maßnahmen

Funktion	Status Nord	Status Süd	Bemerkungen
Korrektheit und Vollständigkeit der Ansible Konfiguration • Server komplett zurücksetzen und Ansible Konfiguration durchführen • läuft durch ohne Fehler	OK	OK	

Einbinden Netzlaufwerk

- Windows 8.1+ Client
- AD Anmeldung
- Transport via AES

Daten schreiben auf Freigabe

- Linux (Samba 4.x) Client
- Transport via AES

Daten lesen von Freigabe

- Windows 8.1+ Client
- Transport via AES

Einbinden Netzlaufwerk "Jeder"

- Linux (Samba 4.x) Client

Schreiben auf Jeder Freigabe

- Windows 8.1+ Client
- AD Anmeldung
- Transport via AES

Quotas

- Windows 8.1+ Client
- AD Anmeldung
- Schreiben von >100 MB

Quotas Jeder

- Linux (Samba 4.x) Client
- Schreiben von >100 MB

Virus schreiben

- EICAR-Testdatei auf Freigabe schreiben
- Eintrag von ClamAV im lokalen Log: Eicar-Test-Signature
- Eintrag im rsyslog auf anderem Server
- Zugriff wird verhindert

Zugriff via ipv6 nicht möglich

- Zugriff von interner IP auf SMB Freigabe per IPv6
- Zugriff von externer IP auf SMB Freigabe per IPv6

Automatisches Backup

- anlegen von Referenzdatei auf persönlicher Freigabe
- anlegen von Referenzdatei auf Jeder-Freigabe
- Überprüfung der Backup-Datei auf dem jeweils anderen Server nach 24h

---

Syslog

- Anmeldung mit Admin-Account auf Server
- Ausführen von SUDO-ping
- Anmeldung und ping sind im lokalen Log vorhanden
- Anmeldung und ping sind im remote-log vorhanden

Gesicherter Login

- Test-Login mit Admin-Account und falschem Passwort wird nach max. 10 Versuchen abgelehnt
- Zeitpunkt des letzten Login und Logout werden beim Anmelden mitgeteilt