Inhalt

[Teil 1 3](#_Toc351374253)

[Aufgabenstellung 3](#_Toc351374254)

[ANFORDERUNGEN 3](#_Toc351374255)

[Evaluation 3](#_Toc351374256)

[Abnahme der Evaluation 3](#_Toc351374257)

[Use Case 3](#_Toc351374258)

[Usability 3](#_Toc351374259)

[Performance 4](#_Toc351374260)

[Datensicherheit 4](#_Toc351374261)

[Betrieb und Nachhaltigkeit 4](#_Toc351374262)

[Quantitative Anforderungen 4](#_Toc351374263)

[Server 4](#_Toc351374264)

[Versionsverwaltung 4](#_Toc351374265)

[Userdokumentation 4](#_Toc351374266)

[Vorkenntnisse 5](#_Toc351374267)

[Vorarbeiten 5](#_Toc351374268)

[Firmenstandards 5](#_Toc351374269)

[Projektorganisation 5](#_Toc351374270)

[Zeitplan 6](#_Toc351374271)

[Arbeitsjournal 7](#_Toc351374272)

[Teil 2 17](#_Toc351374273)

[Management Summary 18](#_Toc351374274)

[Ausgangssituation 18](#_Toc351374275)

[Umsetzung 18](#_Toc351374276)

[Erwartetes Ergebnis 18](#_Toc351374277)

[Analysieren 18](#_Toc351374278)

[Evaluation 18](#_Toc351374279)

[Eigene Empfehlung 21](#_Toc351374280)

[Ausgewählte Software 21](#_Toc351374281)

[Planen 21](#_Toc351374282)

[Konzept 21](#_Toc351374283)

[Realisieren 23](#_Toc351374284)

[Installation 23](#_Toc351374285)

[Konfigurieren 26](#_Toc351374286)

[Benutzerdokumentation 35](#_Toc351374287)

[Wie funktioniert es? 35](#_Toc351374288)

[Testen 36](#_Toc351374289)

[Abgabe 36](#_Toc351374290)

[Anhang 37](#_Toc351374291)

[Abkürzungen / Fachbegriffe 37](#_Toc351374292)

[Apache2 37](#_Toc351374293)

[PHP5 37](#_Toc351374294)

[PostgreSQL 37](#_Toc351374295)

[SimpleSAMLphp 37](#_Toc351374296)

[Filesender 37](#_Toc351374297)

# Teil 1

## Aufgabenstellung

ANFORDERUNGEN  
Aufgabe der IPA ist es das Pflichtenheft des Auftraggebers (Gruppenleiter IT-Betrieb) welches mit der folgenden Aufgabenstellung kompatibel ist, zu erfüllen.  
Der neue Web-Dienst soll am Ende der IPA unseren Kunden zur Verfügung stehen und unser Dienstleistungsportfolio erweitern.

Evaluation  
Eine kurze, aussagekräftige Evaluation macht klar, aus welchem Grund genau dieses Produkt gewählt wurde. Das heisst, Wahl des Tools muss begründet werden. Der Gruppenleiter muss vor der Umsetzung die Wahl absegnen.

Abnahme der Evaluation  
Die Evaluation wird dem Auftraggeber (Gruppenleiter) präsentiert. Das Produkt wird festgelegt.

Use Case  
Die Applikation soll das Austauschen von grossen Dateien ermöglichen: Ein Mitglied der ZHdK möchte eine grosse Datei einer oder mehreren Personen für eine gewisse Zeit an einem geheimen Ort bereitlegen. Die beziehenden Personen erfahren nach dem Bereitstellen, wie und wie lange sie auf die Datei zugreifen können. Nach Ablauf der zeitlichen Frist steht die Datei nicht mehr zur Verfügung. Die Nutzung der Applikation muss beidseitig möglich sein: Ein Mitglied der ZHdK kann einer Person ausserhalb der ZHdK eine Einladung zum Bereitstellen einer Datei senden. Die "externe" Person lädt dann die auszutauschende Datei hoch, woraufhin die anderen beteiligten Personen informiert werden, wie sie die Datei beziehen können.

Usability  
Die Applikation muss über die aufgeführten Webbrowser (Safari ab V.6, Firefox ab V. 17 ESR, Internet Explorer ab V. 9) zugänglich sein und ohne Silverlight, Flash und Java auskommen. Die Nutzung der Funktionalitäten muss dabei so einfach oder selbsterklärend sein, dass weder eine Schulung noch umfassende Anleitungen für die Nutzung benötigt werden. Bei der Beurteilung der Usability kann von einer typischen Person des administrativen Lehrbereiches ausgegangen werden (Sekretariatsmitarbeitende). Der nötige Informationsaustausch mit den Beteiligten sollte über die gängigen Kommunikationskanäle der ZHdK laufen (E-Mail). Dem Zugriff auf die Datei sollen möglichst keine Barrieren im Weg stehen (möglichst keine oder wenige Authentifizierung, möglichst wenige Klicks). Die Authentifizierung für interne User soll mit dem normalen Mail-login erfolgen (mit oder ohne "@login.itz suffix"). Das bedeutet eine Anbindung an unser Active-Directory, Authentication-Gateway oder AAI. Interne User können externe dazu berechtigen eine Datei hochzuladen.

Performance  
Das Verschieben der auszutauschenden Datei darf nicht durch die Applikation an sich, sondern nur durch das Netzwerk an sich beschränkt werden (Bandbreite, Netzanbindung). Falls das Laden der Datei lange Zeit in Anspruch nimmt, soll der Prozess auch "im Hintergrund" weiterlaufen können.

Datensicherheit  
Die auszutauschenden Daten sollen auf unseren eigenen Storage-Systemen zu liegen kommen.

Betrieb und Nachhaltigkeit  
Die Applikation sollte auf freier Software basieren oder mindestens auf offene Standards zurückgreifen. Weiter muss die Software in der IT-Infrastruktur der ZHdK betrieben werden und auf bestehende Systeme (Storage, Server, Netzwerk) zurückgreifen. Der Betrieb und die Wartung der Applikation sollten möglichst klein sein und kein seltenes Spezialwissen erfordern. Für einen zuverlässigen Betrieb sollte die Applikation in ein automatisiertes Überwachungssystem (Nagios) eingebunden werden können.

Quantitative Anforderungen  
- Grösse einer Datei: maximal 10 Gigabyte  
- Vorhaltezeit der Datei: mindestens 2, maximal 28 Tage.  
- Messgrösse für Usability: Der typische Arbeitsplatz zur Nutzung des Dienstes ist über Kabelverbindung mit 100 Megabit am Netzwerk angeschlossen.

Server  
Die Software muss auf einem Firmen-internen Webserver mit Storage- und Actice-Directory-Anbindung aufgesetzt werden.  
Die Verbindung vom Server zum Actice-Directory muss verschlüsselt erfolgen.  
Die Verbindung vom Client zum Server muss verschlüsselt und mit gültigem Zertifikat erfolgen.  
Aus den detaillierten Log-Dateien können Verbindungsfehler und Autentisierungsfehler zeitgenau Identifiziert werden.

Versionsverwaltung  
Alle bearbeiteten Konfiguartions-Dateien werden in einem firmen-internen Git-Repository verwaltet.  
Alle für die IPA relevanten Dokumente und Dokumentationen werden in einem firmen-internen Git-Repository verwaltet.

Userdokumentation  
Es soll eine vollständige step-by-step Benuterdokumentation erstellt werden.

## Vorkenntnisse

Apache2 = In der Schule kurz angeschaut und schon produktiv eingesetzt aber nicht allzu oft  
Postfix = Grundkenntnisse aus der Schule  
PHP5 = Noch nie richtig eingesetzt  
SimpleSAMLphp = keine Vorkenntnisse   
FileSender = keine Vorkentnisse  
PostgreSQL = Keine Vorkentnisse, dafür MySQL  
Linux = Betreut 2 Linux Server

## Vorarbeiten

Damit ich während der IPA nicht auf bestellte Dienste etc. warten muss, habe ich schon einige Sachen bei mir im Unternehmen vorbestellt.

Debian Linux Server Transporter  
IP-Reservationen 195.176.254.107  
 10.10.10.107  
DNS-Reservation transporter.zhdk.ch  
QuoVadis Server Zertifikat   
NFS-Storage [filer2 (10.10.10.201)] /vol/transporter/transporter  
Firewall Gruppe Webserver

## Firmenstandards

Die Zürcher Hochschule der Künste hat keine richtigen Firmenstandards.   
Die Firmenstandards sind auf Linux Seite „The Debian Way“, was eine andere Bezeichnung für Common Practices ist. Das beinhaltet Daten und Software im richtigen Verzeichnis zu speichern, den Server „sauber“ zu halten und Software wenn möglich mit dem Installer apt-get zu installieren.

## Projektorganisation

Das Projekt wird vom Lehrenden Dominic Näpflin der Zürcher Hochschule der Künste geleitet und ausgeführt. Als Projektmanagementmethode wird ein angepasster Wasserfall verwendet, mit den Phasen Analyse, Planen, Realisieren, Testen und Abgabe.

## Zeitplan



## Arbeitsjournal

| Arbeitstag 1, 05.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:**  Zeitplan erstellen  Evaluation beginnen und beenden | | | **Erreicht**:  Zeitplan erstellt  Evaluations unterlagen erarbeitet | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:**  08:30-12:00  12:30-17:00  17:00-17:30 | **Arbeitsschritte/Teilschritte**  Zeitplan erstellen  Evaluation erstellen  Arbeitsjournal erstellen & nachtragen | **Zuständigkeit:** | **Soll-Termin:**  05.03 12:00  05.03 17:00  05.03 17:30 | **Ist-Termin:**  05.03 12:00  05.03 17:00  05.03 17:25 |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung: Am ersten Tag lief alles wie geplant, ich konnte genug Zeit in den Zeitplan und die Evaluation stecken, damit ich einen guten Zeitplan und richtige Argumente für die Auswahl der Software habe.  Verlauf: Alles verlief wie geplant.  Probleme: Probleme Traten heute keine auf.  Lösungen: | | | Morgen werde ich mein Ergebnis der Evaluation dem Teamleiter Präsentieren und mit ihm entscheiden, welche Software eingesetzt wird. | |

| Arbeitstag 2, 06.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:**  Gespräch mit dem Experten  Evaluation präsentieren  Konzept erarbeiten | | | **Erreicht:**  Gespräch gehalten  Evaluation präsentiert  Evaluation besprochen  Konzept erstellt | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:**  07:30-08:00  08:00-08:45  08:45-09:10  09:10-09:50  09:50-12:00  12:30-16:30 | **Arbeitsschritte/Teilschritte**  Auf Expertengespräch vorbereitet  Gespräch mit Expert  Vorbereitung auf Evaluationspräsentation  Evaluationspräsentation  Evaluation Nachbesprechung  Konzept erstellen | **Zuständigkeit:** | **Soll-Termin:**  08:00-09:00  09:00-10:00  10:00-12:00  12:30-16:30 | **Ist-Termin:**  8:00-08:45  09:00-10:00  10:00-12:00  12:30-16:30 |
| **Reflexion** | | | **Planung für die Schultage und das Wochenende** | |
| Allgemeine Zusammenfassung: Heute musste ich viel schreiben, was mir eigentlich nicht sehr liegt. Aber ich bin trotzdem gut vorangekommen und konnte alle gesetzten Ziele erreichen. Auch das Evaluationsgespräch war sehr hilfreich, da wir dort nochmal die Ziele festlegten die nicht so genau definiert waren.  Verlauf: Das Gespräch mit dem Experten und das Evaluationsgespräch verliefen meiner Meinung nach sehr gut und war auch informativ. Der restliche Tag verlief ziemlich so wie ich es geplant hatt. | | | Ich werde diese Woche noch nicht neben der normalen Arbeitszeit arbeiten, da ich sehr gut im Zeitplan liege. | |

| Arbeitstag 3, 11.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:**  Servereinstellungen überprüfen  Server Updaten  Serverdienste Installieren/konfigurieren (PHP5, Apache2, SimpleSAMLphp, postfix)  Installation dokumentieren | | | **Erreicht:**  Servereinstellungen überprüft  Server war auf dem neusten Stand  Serverdienste installiert & konfiguriert  Installation dokumentiert | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:**  09:00-10:00  10:00-12:50  13:20-17:30  09:00-17:30 | **Arbeitsschritte/Teilschritte**  Server Einstellungen überprüfen  Serverdienste installieren  Serverdienste konfigurieren  Installation/Konfiguration dokumentieren | **Zuständigkeit:** | **Soll-Termin:**  09:00-11:00  12:00-17:30  12:00-17:30  09:00-17:30 | **Ist-Termin:**  09:00-10:00  10:00-12:50  13:20-17:30  09:00-17:30 |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung: Die Servereinstellungen waren alle korrekt und Updates waren keine nötig. So konnte ich viel Zeit sparen und diese in die Installation und Konfiguration der Dienste einsetzen. Dokumentiert habe ich, aber noch nicht so ausführlich wie es am Schluss sein soll.  Verlauf: Die Einstellungen des Server zu kontrollieren ging viel schneller als erwartet und die Updates auch, da sie ja schon auf dem aktuellsten Stand sind.  Probleme: Ich hatte einige Probleme mit den Apache Konfigurationsfiles, es wollte einfach nicht funktionieren.  Lösungen: Ich suchte im Internet nach einer Lösung und fand eine die für meine Einstellungen funktioniert hat. | | | Morgen werde ich die Server Zertifikate installieren und den apache so konfigurieren das er sie akzeptiert. Ausserdem werde ich die Software Filesender herunterladen und installieren. | |

| Arbeitstag 4, 12.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:**  Serverzertifikat einbinden  PostgreSQL installiert und konfiguriert  FileSender Herunterladen  Installation dokumentieren | | | **Erreicht:**  Serverzertifikat eingebunden  PostgreSQL installiert & konfiguriert  FileSender heruntergeladen  Installation dokumentiert | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:**  08:30-11:30  11:30-13:00  13:30-17:30  08:30-17:30 | **Arbeitsschritte/Teilschritte**  Serverzertifikat einbinden  Filesender Herunterladen  PostgreSQL installieren und konfiguriert  Installationsdokumentation nachtragen | **Zuständigkeit:** | **Soll-Termin:**  08:30-10:30  10:30-13:00  13:30-17:30  08:30-17:30 | **Ist-Termin:**  08:30-11:30  11:30-13:00  13:30-17:30  08:30-17:30 |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung: Heute hatte ich einige Probleme mit dem Zertifikat, ich konnte aber alle Probleme lösen und ich bin gut vorangekommen.  Verlauf: Weil ich für das Zertifikat länger hatte als geplant, musste ich mich ein bisschen ranhalten. Mir ist ausserdem aufgefallen, dass ich PostgreSQL vor dem Filesender installieren muss, damit ich sie gleich einbinden kann, deswegen habe ich diese Aufgabe vorgeschoben  Probleme: Ich konnte das Zertifikat und dem Schlüsse nicht einbinden. Als erstes wurde das Zertifikat nicht gefunden und danach der Schlüssel nicht akzeptiert.  Lösungen: Das Zertifikat war unter einem falschen Namen in der Apachekonfiguration gespeichert und wurde deswegen nicht gefunden, der Schlüssel war nicht korrekt auf unserem Server gespeichert. Die Formatierung in dem Key-File war nicht korrekt, nach Änderungen funktionierte es ohne Probleme. | | | Morgen werde ich die Installation des Filesenders die ich heute nicht machen konnte nachholen und den Filesender mit unserer AD verbinden. | |

| Arbeitstag 5, 13.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:**  FileSender installieren  AD Anbindung realisieren | | | **Erreicht:**  FileSender ist installiert  AD Anbindung funktoniert | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:**  08:30-13:00  13:30-17:30  08:30-17:30 | **Arbeitsschritte/Teilschritte**  FileSender installiert  AD Anbindung realisiert  Installations- und Konfigurationsdokumentation nachtragen | **Zuständigkeit:** | **Soll-Termin:**  08:30-13:00  13:30-17:30  08:30-17:30 | **Ist-Termin:**  08:30-13:00  13:30-17:30  08:30-17:30 |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung: Die Installation des Filesenders ging recht gut voran, es traten keine grossen Fehler und Ungereimtheiten auf. Leider konnte ich den FileSendern nicht über apt-get installieren, da es ein extra repository gewesen ist und es nicht auf dem neusten Stand war.  Verlauf:  Probleme: Ich musste mich erst in das simpleSAMLphp einlesen, da die AD Anbindung über dieses Tool geregelt ist wird. Ich hatte einige Zeit bis ich endlich aus dem Tool schlau wurde und die Konfiguration in Angriff nehmen konnte.  Lösungen: Nach einigem Lesen im Internet, ist mir klar geworden, wie ich die Konfiguration verändern musste. | | |  | |

| Arbeitstag 6, 18.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:**  NFS-Storage einbinden  SSL/TLS einrichten  Installation und Konfiguration dokumentieren | | | **Erreicht:** | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:** | **Arbeitsschritte/Teilschritte** | **Zuständigkeit:** | **Soll-Termin:** | **Ist-Termin:** |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung:  Verlauf:  Probleme:  Lösungen: | | |  | |

| Arbeitstag 7, 19.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:** | | | **Erreicht:** | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:** | **Arbeitsschritte/Teilschritte** | **Zuständigkeit:** | **Soll-Termin:** | **Ist-Termin:** |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung:  Verlauf:  Probleme:  Lösungen: | | |  | |

| Arbeitstag 8, 20.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:** | | | **Erreicht:** | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:** | **Arbeitsschritte/Teilschritte** | **Zuständigkeit:** | **Soll-Termin:** | **Ist-Termin:** |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung:  Verlauf:  Probleme:  Lösungen: | | |  | |

| Arbeitstag 9, 25.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:** | | | **Erreicht:** | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:** | **Arbeitsschritte/Teilschritte** | **Zuständigkeit:** | **Soll-Termin:** | **Ist-Termin:** |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung:  Verlauf:  Probleme:  Lösungen: | | |  | |

| Arbeitstag 10, 26.03.12 | | | | |
| --- | --- | --- | --- | --- |
| Tagesplanung | | | | |
| **Tagesziele:** | | | **Erreicht:** | |
| **Arbeitsjournal** | | | **Realisierung** | |
| **Zeit:** | **Arbeitsschritte/Teilschritte** | **Zuständigkeit:** | **Soll-Termin:** | **Ist-Termin:** |
| **Reflexion** | | | **Planung für den nächsten Tag** | |
| Allgemeine Zusammenfassung:  Verlauf:  Probleme:  Lösungen: | | |  | |

# Teil 2

## Management Summary

### Ausgangssituation

Wir, die Zürcher Hochschule der Künste, brauchen ein Tool, das unseren Mitarbeitenden erlaubt, grosse Daten zu versenden. Dies funktioniert im Moment nur mit grossem Aufwand und ist deshalb nicht sehr beliebt bei den Mitarbeitenden.

### Umsetzung

Es wird als erstes eine Evaluation gemacht, um die richtige Software zu finden.  
Wenn klar ist, welche Software eingesetzt wird, muss das Konzept so angepasst werden, dass die Anforderungen auch erfüllt werden können. Die Änderungen werden durch den Teamleiter abgesegnet und danach wird mit der Implementierung des Tools begonnen.  
Am Ende werden noch die Funktionen getestet und eine Benutzer-Dokumentation erstellt.

### Erwartetes Ergebnis

Ein Tool soll installiert und konfiguriert werden, mit dem unsere Mitarbeitenden grosse Dateien versenden können, ohne grossen Aufwand zu betreiben. Das Tool soll einfach zu bedienen sein, damit keine grossen Anleitungen von den Mitarbeitenden gelesen werden müssen. Es soll auch möglich sein Daten, die eine gewisse Zeit gespeichert bleiben, an externe Benutzer, die nicht der ZHdK angehören, zu senden. Ausserdem soll es auch möglich sein, externen Benutzern das Recht zu geben, eine Datei über das Tool zu senden.

## Analysieren

Der Abschnitt Analyse behandelt die Evaluation der Software. Es werden verschiedene Möglichkeiten durchgegangen und wird mit dem Teamleiter entschieden, welche eingesetzt wird.

### Evaluation

Weil keine Software explizit genannt wurde, wird eine Evaluation gemacht, die bestimmen soll, welche Software genutzt werden soll. Die Kriterien, welche die Software einhalten muss, werden der Aufgabenstellung entnommen.  
Nach der Evaluation wird das Ergebnis dem Teamleiter präsentiert und mit ihm zusammen entscheiden, welche Software eingesetzt wird.  
Es stehen fünf verschiedene Softwarelösungen zur Evaluation bereit: Filesender, Dropbox, Teamdrive, OwnCloud und Google Apps.

Folgende Kriterien wurden dem Text entnommen:

**User Case Usability**

Daten Bis zu 10 GB versenden Webbrowser kompatibel

AD-Anbindung benötigt keine Plug-Ins

Ablaufdatum der Daten ist Selbsterklärend

Daten an externe Kunden Senden Kommunikation über Email

Externe können Daten hochladen

**Performance Datensicherheit**

Applikation ist schnell Dateien werden auf internen Ablage gespeichert

Hochladen soll im Hintergrund möglich sein Daten werden Verschlüsselt übertragen und gespeichert

**Betrieb & Nachhaltigkeit**

Freie Software Geringe Wartung

Offene Standards Kein Extrawissen notwendig

Läuft im ITZ mit bereits vorhandenen Servern Überwachung möglich

Damit eine gute Einschätzung der Software gemacht werden kann, wird eine Präferenzmatrix erstellt, mit der die Gewichtung der verschiedenen Kriterien besser eingeschätzt werden kann.

Die Präferenzmatrix wird erstellt, indem man die Anforderungen in einer Liste einträgt. Danach werden immer die zwei Anforderungen die untereinander stehen miteinander verglichen und die wichtigere Anforderung wird eine Spalte weiter geschoben. So erhält man eine gute Gewichtung, die nachvollziehbar ist.

Beim Betrachten der Präferenzmatrix scheint es einige sehr wichtige Punkte zu geben und andere, die nicht so wichtig sind. Dieser Eindruck entsteht, weil die Präferenzmatrix sehr strikt ist und so wichtige Kriterien schon am Anfang ausscheiden können. Das passiert, wenn ein noch wichtigeres Kriterium daneben steht.   
Deswegen wird die Präferenzmatrix benutzt, aber in der Präsentation und dem Gespräch mit dem Teamleiter so definiert, dass auch allen wichtigen Kriterien genug Sorge getragen wird.

Mit Hilfe der Präferenzmatrix kann eine Evaluationstabelle erstellt werden, in der die verschiedenen Anforderungen mit der Gewichtung dargestellt werden. So entsteht eine aussagekräftige Tabelle, die hilft, die richtige Software für unsere Anforderungen zu finden.



Das Ergebnis der Evaluationstabelle zeigt klar, dass drei der fünf Software-Lösungen in Frage kommen: Filesender 99,4, OwnCloud 98.5 und Teamdrive 95.9. Aber jede hat Vor- und Nachteile, die für den Entscheid mit dem Teamleiter besprochen werden müssen um abzuklären.

#### Vor- und Nachteile

**FileSender** kann Daten mit einem Link einfach an interne, aber auch externe User/Email-Adressen versenden. Ausserdem ist es möglich, externen Usern einen temporären Account zu erstellen, mit dem Sie Daten an ein ZHdK-Mitglied senden können.   
Der Nachteil dieser Software ist, dass SimpleSAMLphp eingesetzt wird. Dies ist ein Tool, das keiner unserer Systemadministratoren kennt.

**OwnCloud** ist ein sehr grosses Programm und erinnert stark an Dropbox. Mit diesem Programm lassen sich Daten in der Cloud gut verwalten. Auch bei dieser Software ist es möglich, Daten an externe Benutzer zu senden. Leider kann OwnCloud aber keine temporären User erstellen und ist für unseren Zweck überdimensioniert.

**Teamdrive** ist ähnlich wie OwnCloud, aber kostenpflichtig. Dies bedeutet, man zahlt für die Software und deren Benutzung pro Monat einen Betrag an die Entwicklerfirma. Der Vorteil von Teamdrive ist, dass ein guter Support geboten wird und dass eine starke Verschlüsslung der Dateien möglich ist.

### Eigene Empfehlung

Ich empfehle nach der Evaluation, die Software Filesender, da sie genau das macht, was wir wollen. Dies zeigt auch die Evaluationstabelle mit der fast perfekten Gewichtung von 99.4.

Das einzige Problem ist, dass niemand sich mit SimpleSAMLphp auskennt. Dies ist eine Software, die es ermöglicht, einen Single Singe On Punkt zu erstellen. Weil wir das aber nicht benutzen und brauchen, ist die zusätzliche Komplexität nicht ideal. Dazu kommt noch, dass bei grossen Änderungen ein grösserer Aufwand entstehen könnte, als bei anderen Lösungen, die den Systemadministratoren bekannt sind.

Trotzdem denke ich, dass diese Software die richtige ist.

### Ausgewählte Software

Wir haben uns für die Software Filesender entschieden, weil einige wichtige Kriterien nur von dieser Software erfüllt werden.   
Der Hauptgrund für den Entscheid gab das Kriterium „Externe können Daten hochladen“ und dass die Software genau das kann, was wir wollen. Es kann nicht zu einem weiteren Ablageplatz für Daten werden, die nie gelöscht werden.  
Wir haben beschlossen, dass einige Änderungen in den Anforderungen gemacht werden müssen. Diese werden im Konzept erläutert

## Planen

In dieser Phase wird ein Konzept mit detaillierten Angaben für den Transporter-Dienst erstellt.

### Konzept

Das Konzept aus dem Pflichtenheft wird fast 1 zu 1 übernommen. Es müssen nur einige Veränderungen an der Aufgabenstellung gemacht werden, damit es mit der ausgewählten Software auch umsetzbar ist.

#### Anforderungen der Zielgruppe

Es soll mit der Software Filesender, eine Webseite aufgebaut werden, mit der ZHdK-Kunden (ZHdK-Login) Daten an externe und interne Benutzer senden können. Die Daten müssen auf unseren internen Server gespeichert und der Link zu den Daten muss mittels E-Mail versendet werden können.

#### Zu erstellende Dokumente

Neben der funktionierenden Software muss auch eine detaillierte Installations- und Konfigurationsdokumentation erstellt werden, damit jeder Systemadministrator der ZHdK die Installation und Konfiguration durchführen kann. Diese Anleitungen werden in der internen Wikipedia gespeichert und zusätzlich dem IPA-Bericht beigelegt.  
Auch eine Step-by-Step Anleitung für die Benutzer soll erstellt werden, die dann in das Services-Portal des ITZ zugänglich gemacht wird.

#### Geänderte Anforderungen

Weil einige Anforderungen nicht oder nur in abgewandelter Form von FileSender unterstützt werden, mussten die Anforderungen umgeschreiben und klarer definiert werden.

Die Software muss nicht mehr selbsterklärend sein, da diese Formulierung nicht klar definierbar ist und von Person zu Person unterschiedlich ist. Die neue Formulierung ist wie folgt:  
Eine Person, die Erfahrung hat, mit einem Web-Browser im Internet zu navigieren, kann mit Hilfe der Step-by-Step-Anleitung alle Funktionen des Dienstes nutzen.

Auch die Angaben über das Authentifizieren und Klicken mussten ein wenig anpasst werden, da nicht klar definiert ist, was „wenig Klicks“ sind und was mit einer einfachen Authentifizierung gemeint ist.   
Die Anzahl der erlaubten Klicks für Einloggen, Datei hochladen und Versenden ist auf 10 gesetzt und das Authentifizieren soll beim hochladen der Datei nur einmal erforderlich sein.

Weil sich Filesender klar als Sieger platzieren konnte und diese eine spezielle Software zum Einloggen benutzt, die niemand an der ZHdK kennt, mussten wir die Angabe „keine Spezialwissen notwendig“ auf „sehr gute Installations- und Konfigurationsdokumentation“ ändern.

Um die Effizienz der Software zu testen, wird die Geschwindigkeit des Hoch- und Herunterladen mit unserem FTP-Server und mit der VPN-Verbindung verglichen. Genauere Angaben zum Test der Performance werden im Abschnitt Testen angegeben.

Das Hoch- und Herunterladen von Daten soll in einem separaten Tab funktionieren.

#### Backup

Für diesen Webdienst wird kein Backup eingerichtet. Die Dateien liegen auf dem NFS-Storage und werden dort nicht noch zusätzlich gesichert, da es nur ein Tool zum übertragen von Daten ist und somit diese Daten noch irgendwo anders gespeichert sind.   
Die Konfiguration hingegen liegt lokal auf dem virtuellen Server und wird automatisch mit den anderen virtuellen Servern gesichert.

#### Vorgehen

Auf dem vorbereiteten Linux-Debian-Server werden die Software Filesender und weitere benötigten Applikationen installiert und konfiguriert, damit es produktiv einsetzbar ist.

Als erstes wird der Server so vorbereitet, dass mit der Installation der Software und den nötigen Konfigurationen begonnen werden kann.   
Dazu zählen:

* QuoVadis Root Zertifikat einbinden

Linux OS updaten/upgraden falls nötig

Nach den Vorbereitungen wird mit der Installation der Software begonnen. Dazu werden Anleitungen aus dem Internet für unseren eigenen Zweck umgeschrieben.

Die Anbindung von FileSender an unsere Active-Directory wird über das Tool SimpleSAMLphp realisiert.

Als Datenbank wird PostgreSQL verwendet, da dieser Datenbanktyp der Standard von FileSender ist. Weil wir kein PostgreSQL-Server haben, wird die Datenbank lokal auf dem Transporter installiert und betrieben.

Als Webdienst wird Apache2 mit PHP-Modulen verwendet und so eingestellt, dass der Dienst nur über HTTPS erreichbar ist. Dazu wird ein Zertifikat der Zertifizierungsstelle QuoVadis verwendet.

Den NFS-Container wird so eingebunden, dass nur die Daten, die hochgeladen werden, im NFS-Speicher abgelegt werden.

## Realisieren

### I**nstallation**

Alle Einstellungen und Konfigurationen, die in dieser Installationsdokumentation gemacht werden, sind auf dem Server transporter.zhdk.ch durchzuführen. Es ist ein Linux Debian 6.0 Server, der im ZHdK-Cluster ist. Der Server hat einen Intel(R) Xeon(R) CPU L5640 @ 2.27GHz Prozessor, 6 GB RAM und 15 GB interner Speicher.  
Aus Platzgründen werden hier nur die Teile aus den Konfigurationsdateien aufgelistet, die ich selber verändert habe und relevant sin. Die gesamten Konfigurationsdateien befinden sich im Anhang.

#### Updates

Zuerst muss überprüft werden, ob das Debian Betriebssystem auf dem aktuellen Stand ist. Dies funktioniert mit folgenden Befehlen:

* apt-get update
* apt-get upgrade

#### Root Zertifikat

Als zweiter Schritt muss das QuoVadis Root Zertifikat auf dem Debian Server installiert werden. Da QuoVadis den Zeitpunkt verpasst hat, bei den Debian Entwicklern ihr Root Zertifikat rechtzeitig einzureichen, ist es in der Version 6.0 nicht standardmässig installiert.

Das Root Zertifikat kann von einem produktiven Server (hier wiki.zhdk.ch) bei dem es schon installiert ist, kopiert werden. Der Befehl kopiert das Zertifikat und speichert es auf dem Transporter in das Verzeichnis /etc/ssl/certs/.

scp dominic@wiki.zhdk.ch:/etc/ssl/certs/QuoVadis\_Global\_SSL\_ICA.pem /etc/ssl/certs/

Nun muss das Zertifikat in die Datei /etc/ldap/ldap.conf eingetragen werden

# Hier wird das QuoVadis Root Zertifikat im ldap eingebunden  
TLS\_CACERT /etc/ssl/certs/QuoVadis\_Global\_SSL\_ICA.pem

Um zu testen, ob das Root-Zertifikat korrekt eingebunden wurde, kann mit dem Befehl ldapsearch versucht werden, mit dem Server zu verbinden. Zuerst muss aber noch das Packet ldap-utils installiert werden.

* apt-get install ldap-utils
* ldapsearch -Z -H "ldap://172.30.10.12" -x -D 'CN=ldapsearch,OU=Service Accounts,OU=Accounts,OU=\_ZHdK manuell,DC=vera,DC=hgka,DC=ch' -w '<Passwort>' -b 'dc=vera,dc=hgka,dc=ch' '(msNPAllowDialin=FALSE)'

#### Apache2

Als nächstes muss der Webdienst installiert werden. Wir haben uns für Apache2 entschieden, da alle unsere internen Webserver mit Apache2 laufen.

apt-get install apache2

Da die Verbindung zum Webserver verschlüsselt sein soll, muss das apache2 SSL-Modul und die SSL-Seite aktiviert werden.

* a2enmod ssl
* a2ensite default-ssl
* /etc/init.d/apache2 restart

#### PHP5

Nun muss noch PHP5, ein Apache2 PHP-Modul und das PHP-LDAP-Modul installiert werden.

* apt-get install php5 libapache2-mod-php5 php5-ldap

#### PostgreSQL

Die PostgreSLQ Datenbank wird mit folgendem Befehl erstellt:

* apt-get install postgresql php5-pgsql

Damit vom localhost auf die PostgreSQL-Datenbank eingeloggt werden, kann muss noch die Konfigurationsdatei /etc/postgresql/8.4/main/pg\_hba.conf angepasst werden:

# Database administrative login by UNIX sockets  
local all postgres ident  
# TYPE  DATABASE USER CIDR-ADDRESS METHOD  
# "local" is for Unix domain socket connections only  
local all all ident  
# IPv4 local connections:  
host all all 127.0.0.1/32 **md5**  
# IPv6 local connections:  
host all all ::1/128 **md5**

Nach den Änderungen an der Konfigurationsdatei muss der Dienst neu gestartet werden.

* /etc/init.d/postgresql restart

#### SimpleSAMLphp

Eigentlich könnte dieses Packet über den Befehl „apt-get install“ installiert werden. Aber diese Version ist leider nicht die aktuelle Version, die von den FileSender-Programmierer verwendet wird. Daher wird das Packet von der site googlecode.com heruntergeladen und von Hand installiert.

Zuerst wird in den /tmp Ordner gewechselt und danach kann mit dem wget-Befehl das Packet heruntergeladen werden:

* cd /tmp
* wget <http://simplesamlphp.googlecode.com/files/simplesamlphp-1.10.0.tar.gz>

Nach dem Herunterladen muss überprüft werden, ob das Packet richtig und vollständig gespeichert wurde. Dies funktioniert mit dem Befehl sha1sum.

* sha1sum -b simplesamlphp-1.10.0.tar.gz

Wenn die SHA1 Checksum gleich ist, wie die auf der Download-Page (<http://code.google.com/p/simplesamlphp/downloads/detail?name=simplesamlphp-1.10.0.tar.gz&can=2&q>=), kann das Packet entpackt werden.

Zuerst muss aber für alle FileSender-relevanten Dateien das Verzeichnis(/usr/local/filesender/) erstellt werden.

* mkdir /usr/local/filesender
* cd /usr/local/filesender
* tar -xvf simplesamlphp-1.10.0.tar.gz

Damit der SimpleSAMLphp-Dienst einfach angesprochen werden kann, wird noch ein Symlink erstellt.

* ln -s simplesamlphp-1.10.0/ simplesaml

#### Filesender

Als letztes wird nun FileSender installiert. Dieses Paket wird wie das SimpleSAMLphp-Packet mit dem wget-Befehl heruntergeladen und installiert. Dies muss gemacht werden, weil das Filesender-Packet nicht von den Debian-Repositorien heruntergeladen werden kann.

* cd /tmp/
* wget --no-check-certificate <http://download.filesender.org/filesender-1.5.tar.gz>

Um zu überprüfen, ob das Packet richtig und vollständig heruntergeladen wurde, muss wieder die SHA1 Checksum überprüft werden. Die SHA1 Checksum findet sich unter folgendem Link:  
https://www.assembla.com/wiki/show/file\_sender/Release\_1-5

* sha1sum -b filesender-1.5.tar.gz

Jetzt muss das Packet noch entpackt und an den richtigen Ort verschoben werden.

* mv filesender-1.5.tar.gz /usr/local/filesender/
* cd /usr/local/filesender
* tar -xvzf filesender-1.5.tar.gz
* mv filesender-1.5/\* /usr/local/filesender/

Nun muss noch die Konfigurationsdatei erstellt werden. Dies wird mit der existierenden .dist Konfiguration gemacht.

* cp config/config-dist.php config/config.php

#### GitHub

Als Versionisierungstool wird GitHub verwendet. Die Installation wird wieder mit apt-get gemacht.

* apt-get install git

### Konfigurieren

Jetzt sind alle notwendigen Pakete installiert, und es kann mit der Konfiguration begonnen werden.

#### NFS-Storage

Weil die Daten, die mit dem Transporter versendet werden, auf einem NFS-Storage liegen sollen, muss dieser nun entsprechend so eingerichtet werden. Die Files des Filesenders werden im Verzeichnis /usr/local/filesender/filesender/files gespeichert. Somit kann man den NFS-Storage auf diesen Ordner mounten und alle Daten werden automatisch auf den NFS-Storage gespeichert. Diese Einstellung wird in der Datei /etc/fstab mit folgendem Eintrag vorgenommen:

10.10.10.201:/vol/transporter/transporter /usr/local/filesender/filesender/file nfs rsize=8192,wsize=8192,nolock,auto 0 0

Danach muss der Mount-Befehl durchgeführt werden.

* mount –a

#### Postfix

Weil Postfix schon standardmässig auf den Debian Server installiert ist, muss nur noch eine kleine Einstellung vorgenommen werden. Postfix soll über den SMTP der ZHdK Mails versenden, denn ohne den SMTP würden die meisten Mails wohl in einem Spam-Ordner landen.   
Dazu müssen in der Postfix-Konfigurationsdatei einige Einstellungen geändert werden:

/etc/postfix/main.cf

# Hier wird die Domain angegeben  
mydomain = zhdk.ch  
#myhostname = transporter.zhdk.ch  
# Hier wird angegeben, von welchen Quellen Postfix Mails annehmen soll  
mydestination = $myhostname, localhost, localhost.localdomain, localhost  
# Hier wird der SMTP angegen der verwendet werden soll  
relayhost = SMTP.zhdk.ch

#### Filesender

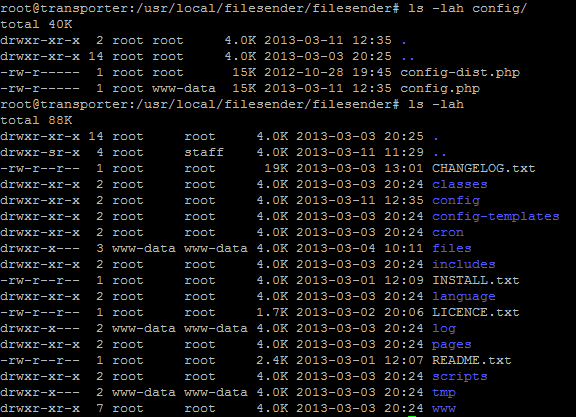
Beim Filesender muss als erstes die Rechtevergabe geregelt werden, denn einige Konfigurationsdateien und Verzeichnisse müssen lesbar sein, andere hingegen nicht.   
Die Verzeichnisse Files/, log/ und tmp/ müssen dem Benutzer und der Gruppe www-data gehören.

* chown www-data:www-data tmp files log
* chown o-rwx tmp files log
* chgrp www-data config/config.php
* chmod o-rwx tmp/ files/ log/
* chmod g-w tmp/ files/ log/

Die Konfigurationsdatei /usr/local/filesender/filesender/config/config.php muss noch der Gruppe www-data zugewiesen werden.

* chown root:www-data config/config.php

Die Rechte der Verzeichnisse und Daten müssen wie auf dem folgenden Screenshot aussehen.



Weil die temporären Daten auch sehr gross werden können, wird der Ordner tmp/ in den Ordner files verschoben, damit alle Dateien, die beim Hochladen entstehen, auf dem NFS-Storage liegen.

* mv /usr/local/filesender/filesender/tmp /usr/local/filesender/filesender/files/

Die maximale Upload-Grösse muss auch noch in der Datei /usr/local/filesender/filesender/config.php geändert werden.

// Hier wird angegeben, das mit Scrips bis zu 10 GB grosse Files hochgeladen werden können  
$config['max\_flash\_upload\_size'] = '10737418240';

// Hier wird angegeben, das mit HTML5 bis zu 10 GB grosse Files hochgeladen werden können  
$config['max\_html5\_upload\_size'] = '10737418240';

//Wenn die Daten mit HTML5 hochgeladen werden, wird ein chunk verwendet, wenn dieser vergrössert wird, kann die Übertragungsrate verbessern.  
$config["upload\_chunk\_size"] = '2000000';/

Weitere Einstellungen müssen auch noch in der Konfigurations-Datei config.php gemacht werden.

$config['admin'] = ''; // UID's (from $config['saml\_uid\_attribute']) that have Administrator permissions  
$config['adminEmail'] = 'service.itz@zhdk.ch'; // Email Adresse an die Meldungen gesendet werden  
$config['Default\_TimeZone'] = 'Europe/Berlin'; //Welche Zeitzone wird die Seite verwenden  
$config['site\_defaultlanguage'] = 'en\_AU'; // Welche Sprache wird als Standard verwendet  
$config['site\_name'] = 'Transporter.zhdk.ch FileSender'; // Wie sich der Dienst selber bezeichnet

Da es nachvollziehbar sein soll, wer was Hoch- und Runtergeladen hat, muss folgende Option auf true geändert werden, denn so wird jeder Client protokolliert.

$config["client\_specific\_logging"] = true; // sollen client gelogt werden (true/false)

Einstellungen für die Authentifizierung werden im Abschnitt saml settings gemacht.

$config['saml\_email\_attribute'] = 'mail'; // Attribut, welches die E-Mailadresse angibt  
$config['saml\_name\_attribute'] = 'cn'; // Attribut, welches den Namen des Benutzers angibt  
$config['saml\_uid\_attribute'] = 'sAMAccountName'; // Attribut, welches den Benutzer identifiziert

Unter dem Abschnitt Server settings, kann eingestellt werden wie viele Tage ein File gespeichert werden soll, welche Suffixe nicht erlaubt sind und wieviele E-Mails pro Datei versendet werden kann.

$config['default\_daysvalid'] = 20; // Gibt die maximale Zeit an, die ein File gespeichert wird.  
$config['ban\_extension'] = 'exe,bat'; // Gibt an welche Dateiendungen blockiert werden.  
$config["max\_email\_recipients"] = 100; // Gibt die maximale Anzahl gleichzeitig versendbarer E-Mails an, 0 ist unlimitiert.

Damit nur auf unseren Active-Directory Server verbunden werden kann, muss bei der folgenden Option das selbst erstellte LDAP-Profil stehen.

$config['site\_authenticationSource'] ="zhdk-ldap"; //Gibt den verwendeten SimplaSAMLphp Authentisierungspunkt an

Bei den folgenden Optionen wird angegeben, wo welche Daten gespeichert werden sollen.

$config['site\_filestore'] = '/usr/local/filesender/filesender/files/';  
$config['site\_temp\_filestore'] = '/usr/local/filesender/filesender/files/tmp/';  
$config['site\_simplesamllocation'] = '/usr/local/filesender/simplesamlphp/';  
$config['log\_location'] = '/usr/local/filesender/filesender/log/';

Die folgenden Einstellungen beziehen sich auf die verwendete Datenbank.

$config["db\_type"] = "pgsql";// pgsql or mysql  
$config['db\_host'] = 'localhost';  
$config['db\_database'] = 'filesender';  
$config['db\_port'] = '5432';

// database username and password  
$config['db\_username'] = 'filesender';  
$config['db\_password'] = '<Passwort>';

Jetzt muss noch das Konfigurationsfile des Moduls Sanity Check an den richtigen Ort speichert werden, denn ohne dieses File kann der Funktionstest des FileSenders nicht durchgeführt werden:

* cp simplesaml/modules/sanitycheck/config-templates/config-sanitycheck.php simplesaml/config/config-sanitycheck.php

Es ist auch möglich; den Banner, der auf der Seite angezeigt, wird zu ändern. Dazu braucht es ein Bild im Format 800x60 Pixel.  
Weil die Zürcher Hochschule der Künste ein Corporate Design hat, ist das Erstellen eines solchen Logos nicht einfach. Die Logos müssen von der Grafik & Design-Abteilung der Zürcher Hochschule der Künste kommen. Es gibt aber eine Website mit Anleitungen und Downloads.

Leider war es nicht möglich, das normale Logo der Zürcher Hochschule der Künste zu verwenden, da es nur in 100%-Grösse abgebildet werden darf, das würde 5.6cm x 2.2cm bedeuten. Deswegen wird das Logo Minimalform verwendet, denn dieses darf verkleinert werden.  
Am einfachsten öffnet man ein Bildbearbeitungsprogramm und verkleinert das Logo Minimalform auf eine Höhe von 60 Pixel. Danach wird das Bild auf 800x60 Pixel verkleinert und muss als Banner.png abgespeichert werden.

Als nächstes muss dieses Bild auf den Server Transfer kopiert werden:

* scp banner.png [dominic@transporter.zhdk.ch](mailto:dominic@transporter.zhdk.ch): /usr/local/filesender/filesender/config/banner.png

Jetzt liegt das Bild am richtigen Ort, ist richtig Formatiert und wird automatisch als Banner verwendet.

Vorher  


Filesender 1

Nachher  


Filesender 2

##### Anpassungen der Seite

/usr/local/filesender/filesender/language/de\_DE.php

// Hier kann der Hilfe Text, der auf der Seite angeboten wird verändert werden.  
$lang["\_HELP\_TEXT"] = '';

// Hier kann der Texfuer die About Seite geschrieben werden.  
$lang["\_ABOUT\_TEXT"] = '';

// Hier kann der Text für die Nutzungsbedingungen eingegeben werden.  
$lang["\_AUPTERMS"] = "Nutzungsbedingungen...";

Als letztes muss beim Filesender noch der Cron-Job eingerichtet werden, der alle abgelaufenen Daten löscht. Diesen Cron-Job ist beim Filesender schon mit dabei und kann an den richtigen Ort verschoben werden.

* php -p /usr/local/filesender/filesender/cron/cron.php >/etc/cron.daily/filesender

#### PostgreSQL

Damit PostgreSQL für den Filesender verwendet werden kann, muss als erstes ein Benutzer erstellt werden. Der Benutzer wird einfachheitshalber filesender genannt und das Passwort wird in einer KeePass-Datenbank gespeichert.

* sudo -u postgres createuser -S -D -R -P filesender

Jetzt muss die Datenbank noch erstellt werden; auch die Datenbank wird filesender genannt.

* sudo -u postgres createdb -E UTF8 -O filesender filesender

Wenn die Datenbank steht, muss noch der Inhalt erstellt werden. Dazu hat FileSender ein Skript:

* psql -h localhost filesender filesender < /usr/local/filesender/filesender/scripts/filesender\_db.sql

#### PHP5

Damit grosse Daten versendet werden können, müssen in dem Konfigurationsfile /etc/php5/apache2/php.ini einige kleine Änderungen vorgenommen werden.

;Die Inputtime wird auf unlimitiert gesetzt, damit auch langsamen Leitungen grosse Dateien versendet werden können.  
max\_input\_time = -1

;Den max-Upload wird auf 10240M (10 GB) gesetzt, damit auch grosse Daten versandet werden können.  
upload\_max\_filesize = 10240M

;Hier wird die maximale Grösse für den Post-Upload angegeben. Das ist der max-Upload + POST-Zusätze. Deswegen ist auch etwas grösser als der upload\_max\_filesize.  
post\_max\_size = 10500M

;Hier kann das temporäre Upload-Verzeichnis angegeben werden. Das Verzeichnis muss genug Platz für die temporären Upload-Daten haben.   
upload\_tmp\_dir = /usr/local/filesender/files/tmp/

;Hier wird angegeben, dass das Error-Log eingeschalten ist.  
log\_errors = On

;Hier wird angegeben, in welches Logfile die Error Meldungen geschrieben werden  
error\_log = php\_errors.log

;Gibt an, ob das Cookie nur über das HTTP-Protokoll zugänglich ist oder nicht  
session.cookie\_httponly = On

Nun muss noch im Suhosin – den Security Patch für PHP – die maximale Länge geändert werden.

suhosin.get.max\_value\_length = 2048

Um die ganzen Einstellungen zu aktivieren, muss Apache2 neu gestartet werden.

/etc/init.d/apache2 restart

#### Apache2

In den Apache2-Einstellungen geht es hauptsächlich um die HTTPS-Anbindung und um Umleitung von HTTP auf HTTPS.   
Diese Einstellungen werden in den Dateien im Verzeichnis /etc/apache2/sites-available/ vorgenommen

Als erstes wird die Umleitung von HTTP auf HTTPS eingestellt. Dies wird in der Datei /etc/apache2/sites-available/default gemacht. Dazu wird die Datei auf old.default umbenannt und eine neue mit Namen default erstellt, damit die Standard-Konfiguration nicht verloren geht.

* mv /etc/apache2/sites-available/default /etc/apache2/sites-available/old.default

nano /etc/apache2/sites-available/default

VirtualHost \*:80>

Redirect permanent / https://transporter.zhdk.ch  
</VirtualHost>

Nun muss default-ssl auch umbenannt werden, damit die Datei filesender-ssl auf default-ssl umbenannt werden kann. So kann man sich einige Zeit sparen, da nur wenige Korrekturen vorgenommen werden müssen und der Rest schon in der Datei steht.

* mv /etc/apache2/sites-available/default-ssl /etc/apache2/sites-available/old.default-ssl
* mv /etc/apache2/sites-available/filesender-ssl /etc/apache2/sites-available/default-ssl

##### Zertifikate

Als nächstes muss noch das Server-Zertifikat und der Schlüssel auf den Transporter kopiert werden. Die Zertifikate liegen auf einem unserer Storage-Server. [\\filer\Services\SER\_SUP\_ITZ\2\_Betrieb\22\_Infrastruktur\223\_Server\Certificates\transporter](file:///\\filer\Services\SER_SUP_ITZ\2_Betrieb\22_Infrastruktur\223_Server\Certificates\transporter)

Die Zertifikate werden auf den Transporter gespeichert, indem der Inhalt der Zertifikate per Copy & Paste in neu generierte Files, im richtigen Verzeichnis gespeichert werden.

* nano /etc/ssl/private/transporter.key
* nano /etc/ssl/certs/transporter.crt.pem

##### SSL Einstellungen

Nun muss noch die HTTPS-Seite Konfiguriert werden. Dies kann man entweder tun, indem man eine eigene Konfigurationsdatei erstellt und diese in den Ordner /etc/apache2/sites-available/ speichert oder man nimmt die vorhandene Datei default-ssl/filesender-ssl und bearbeitet diese.

/etc/apache2/sites-available/default-ssl

<IfModule mod\_ssl.c>  
<VirtualHost \_default\_:443>

#Gibt den Namen des Servers an  
 ServerName transporter.zhdk.ch

#Gibt an wer der Admin des Servers ist  
 ServerAdmin webmaster@localhost

#Gibt an wo das Hauptverzeichnis liegt  
 DocumentRoot /usr/local/filesender/filesender/www  
 <Directory />  
 Options FollowSymLinks  
 AllowOverride None  
 </Directory>

Alias /filesender /usr/local/filesender/filesender/www  
 <Directory "/usr/local/filesender/filesender/">  
 Options FollowSymLinks MultiViews  
 DirectoryIndex index.php  
 AllowOverride None  
 Order deny,allow  
 Allow from all  
 </Directory>

Alias /simplesaml /usr/local/filesender/simplesaml/www  
 <Directory "/usr/local/filesender/simplesaml/www">  
 AllowOverride None  
 Options FollowSymLinks MultiViews  
 Order deny,allow  
 Allow from all  
 </Directory>

#Gibt an, wo das Errorlog gespeichert werden soll  
 ErrorLog ${APACHE\_LOG\_DIR}/error.log  
 CustomLog ${APACHE\_LOG\_DIR}/ssl\_access.log combined

#Gibt an wie geloggt werden soll  
 LogLevel warn

SSLEngine on

#Gibt an wo das verwendete Zertifikat und der Schlüssel liegen  
 SSLCertificateFile /etc/ssl/certs/transporter.zhdk.ch.crt.pem  
 SSLCertificateKeyFile /etc/ssl/private/transporter.key  
</VirtualHost>  
</IfModule>

Wenn die Datei fertig konfiguriert ist, muss sie noch aktiviert werden, falls nicht default-ssl verwendet wurde, denn diese wurde weiter oben schon aktiviert.

Damit die Konfiguration angewendet wird, muss noch der Apache2 neu gestartet werden.

* /etc/init.d/apache2 restart

#### SimpleSAMLphp

Als letztes muss noch die Anbindung an das Active-Directory erfolgen. Dies wird mit dem Tool SimpleSAMLphp realisiert.

Als erstes braucht SimpleSAMLphp einen Salt, um - wenn nötig - einen geheimen Hash zu erstellen. Ein Salt kann mit folgendem Befehl erstellt werden:

* tr -c -d '0123456789abcdefghijklmnopqrstuvwxyz' </dev/urandom | dd bs=32 count=1 2>/dev/null;echo

Die Einstellungen des SimpleSAMLphp sind in verschiedenen Dateien gespeichert. Die grundlegenden Einstellungen sind in der Datei config.php gespeichert.

simplesaml/config/config.php

//Hier wird der Salt eingetragen  
'secretsalt' => 'ut71eaqblv291djmp9mcd723ctjauxuc',

//Bei dieser Option wird das Passwort des Admins bestimmt  
'auth.adminpassword' => '5iMp13:P4s5',

//Hier wird festgelegt, wer bei technischen Problemen kontaktiert werden soll  
'technicalcontact\_name' => 'Dominic Naepflin',  
'technicalcontact\_email' => 'webmaster@zhdk.ch',

Die ganzen LDAP-Einstellungen werden nicht in der Datei config.php gespeichert, sondern in einer separaten Authentisierung-Konfigurationsdatei.

simplesaml/config/authsources.php

//Gibt den Namen der LDAP-Konfiguration an  
'zhdk-ldap'=>array(  
 'ldap:LDAP',  
//Gibt an, auf welchen LDAP-Server verbindet werden soll.  
 'hostname'=>'ldap.zhdk.ch',  
//Gibt an, ob TLS/SSL verwendet werden soll  
 'enable\_tls' => TRUE,  
//Der Pfad zu den Benutzern. Ist deaktiviert, da nicht benötigt, wenn die Suche aktiv ist.  
 //'dnpattern' => 'sAMAccountName=%username%,dc=vera,dc=hgka,dc=ch',  
//Anstelle eines Pfades zu den Benutzern, ist hier die Möglichkeit die Benutzer zu suchen.  
 'search.enable' => TRUE,  
//Gibt die Suchbasis an  
 'search.base' => 'dc=vera,dc=hgka,dc=ch',  
//Welche Attribute sind mit dem Username identisch und sollen bei der Suche benutzt werden.  
 'search.attributes' => array('sAMAccountName', 'mail'),  
//Gibt den Benutzer und das Passwort für den Suchvorgang an   
 'search.username' => 'ldapsearch',  
 'search.password' => '<Passwort>',  
…  
)

### Benutzerdokumentation

#### Für was ist diese Seite gut?

Die Seite Transporter.zhdk.ch ist dazu gedacht kleine und grosse Dateien an externe oder interne Benutzer zu senden.  
Es ist auch möglich, externen Benutzern einen temporären Benutzer zu erstellen, mit dem sie Dateien hochladen und versenden können.

Max. Upload Grösse: 10GB  
Max. Ablaufzeit für Dateien: 20 Tage  
Max. E-Mail Empfänger: 100

### Wie funktioniert es?

Als erstes müssen sie sich auf die Seite transporter.zhdk.ch verbinden. Auf der Seite müssen sie auf den Button „Einloggen“ drücken um auf die Seite zu gelangen auf der Sie sich einloggen können.

Auf dieser Seite müssen Sie dann ihr Benutzername(ohne @login.itz) und ihr Passwort eingeben und auf „Anmelden“ drücken.

Dann werden sie auf eine weitere Seite weitergeleitet wo sie auswählen können was sich machen wollen.  
Datei senden: Sie können eine Datei hochladen und mit einem E-Mail versenden.



Filesender 3

Der Aufbau der Benutzeroberfläche ist ähnlich wie ein E-Mail Formular. Das ist so gewollt, da es auch eine E-Mail versendet wenn alles richtig eingetragen ist. Die vier Punkte Beschreiben genau was gemacht werden muss um eine Datei via E-Mail zu versenden.

Gast einladen: Sie können einem externen Benutzer ein temporäres Konto erstellen, mit dem er/sie auch Dateien hochladen und versenden kann.



Filesender 4

Ein Gast, also ein temporärer Benutzer wird im Filesender oft auch Voucher genannt.

Meine Medien: Hier werden ihnen ihre Dateien angezeigt die noch auf dem Server gespeichert sind.

Auf dieser Seite ist das Formular um Dateien Hochzuladen und sie zu versenden.

## Testen

## Abgabe

**Quellen**

Konzept

<http://www.indoition.com/de/services/technische-dokumentation-software-erstellen.htm>  
<http://anleitung-tipps.anleiter.de/wie-kann-man-ein-konzept-schreiben-vorlage>  
<http://www.access-im-unternehmen.de/index1.php?id=300&BeitragID=263>

Installationsdokumentation

Filesender

<https://www.assembla.com/spaces/file_sender/wiki/Installation_-_Linux_Source>  
<https://www.assembla.com/wiki/show/file_sender/Administrator_reference_manual>

Apache2

[http://httpd.apache.org/docs/2.2/mod/core.html#](http://httpd.apache.org/docs/2.2/mod/core.html)

SimpleSAMLphp

<http://simplesamlphp.org/docs/1.8/ldap:ldap>  
<http://www.worldgoneweb.com/2013/installing-simplesamlphp-and-use-it-as-sp-and-idp-for-development-env-only/>

PHP5

<http://www.isp-star.at/index.asp?file=php-upload-files.asp>  
<http://www.hardened-php.net/suhosin/configuration.html#suhosin.post.max_value_length>

# Anhang

Im Anhang werde ich die ganzen Konfigurationsdateien und andere Wichtigen Anhänge aufführen.

## Abkürzungen / Fachbegriffe

## Apache2

## PHP5

## PostgreSQL

## SimpleSAMLphp

Salt = Ist eine Bezeichnung einer zufällig gewählten Zeichenfolge, die dazu verwendet wird Passwörter oder Andere Dateien verschlüsselt zu übertragen.

## Filesender