

EINFACHER ZUGANG ZUM X.400 VERBUND ÜBER WEBDAV

- EIN QUELLOFFENES, RUBY-ON-RAILS BASIERTES
MODUL FÜR DEN SICHEREN
GESCHÄFTSDATENAUSTAUSCH

KONZEPT

MARTIN LOHRMANN UND HEINZ W. WERTGES
FACHBEREICH DCSM, FACHHOCHSCHULE WIESBADEN

HANS-PETER MAURER
T-SYSTEMS ENTERPRISE SERVICES GMBH, DARMSTADT

1. SEPTEMBER 2008

Inhaltsverzeichnis

1 Die Situation	2
2 Vorschlag	4
3 Details	5
3.1 Verwaltung	5
3.2 Schnittstelle	6
3.3 Tracking und Tracing	6
3.4 Programmablauf	7
3.5 Voraussetzungen	7
3.6 Kosten	8
4 Zusammenfassung	9

1 Die Situation

Zum automatisierten Geschäftsdatenaustausch wird auch heute noch häufig auf das X.400- Verbundnetz zurückgegriffen, da es bekannte und vor allem vertraglich abgesicherte Übertragungswege nutzt. Es stellt ein vom Internet unabhängiges Netzwerk dar, welches die Verfolgung des Mitteilungsverlaufs bis zum Empfänger zulässt. Allerdings ist die Nutzung mit sehr hohen Volumenkosten für die Datenübertragung sowie hohen Anschaltkosten für die Zugangsnutzung verbunden, wodurch es gerade für kleinere Unternehmen stark an Attraktivität einbüsst.

EDIINT AS2 sollte dem ein Ende bereiten, ermöglicht es doch den Datenaustausch über, in Unternehmen zumeist vorhandene, Internetzugänge. Die Volumenkosten für die Übertragung können mit EDIINT AS2 stark reduziert werden, allerdings arbeitet EDIINT AS2 nach dem Point-To-Point Prinzip und ist bei der Einrichtung sehr komplex, wodurch hohe Einmalkosten bei jeder Anbindung von Geschäftspartnern entstehen. Um den sicheren Betrieb mit EDIINT AS2 zu gewährleisten ist außerdem eine komplexe Infrastruktur mit recht hohen Betriebskosten erforderlich. Dadurch kommt auch dieser Ansatz für kleinere Unternehmen meist nicht in Frage.

In einzelnen Bereichen ist es heute allerdings auch für kleine Unternehmen erforderlich, mit ihren Geschäftspartnern in elektronischen, automatisierten Austausch von Daten zu treten. Auf Grund fehlender eigener Mittel greifen diese dann häufig auf das so genannte Web-EDI zurück, was den Automatisierungsvorteil allerdings nur für den großen Geschäftspartner bringt, der Web-EDI nur als Schnittstelle zu seinen klassischen EDI-Systemen einsetzt. Die kleinen Unternehmen müssen per Hand die Daten in das Webfrontend des WebEDI-Systems einpflegen oder proprietäre Schnittstellen bedienen, die keinem Standard genügen.

EDI-Hub Bisher existiert ein, in einer früheren Diplomarbeit an der FH Wiesbaden entstandenes EDI-Hub [Wer07] , welches es KMUs erleichtern soll, mit seinen Geschäftspartnern in Datenaustausch zu treten. Es basiert auf einem Hub-and-Spoke Modell, bei dem der Hub in gemeinschaftlicher Verwaltung aller Nutzer betrieben werden kann. Es verfügt über ein Webinterface zur Administration und Partnerverwaltung, sowie eine in das Interface integrierte Tracking & Tracing Funktion. Diese ermöglicht es dem

Nutzer jeder Zeit den Status seiner Mitteilungen zu verfolgen, was ihm eine gewisse Kontrolle über den Datenfluss gibt. Jeder Nutzer erhält eine (oder mehrere) Mailbox(en) auf dem Hub, über die seine Kommunikation mit den Partnern abläuft. Der Zugriff auf die Mailbox geschieht mittels sftp. Ausgetauscht werden ausschließlich UN/EDIFACT Daten. Das Hub führt eine Validierung der Dokumente durch und stellt sie hinterher dem Empfänger in seiner Mailbox zu.

MessageGate Das BusinessMail X.400[TS] System der T-Systems wurde jetzt um einen neuen Zugang namens MessageGate erweitert. Bei MessageGate handelt es sich um eine Dateischnittstelle zu einem kundenindividuellen Verzeichnis, das als Übergabepunkt für Mitteilungen genutzt wird. MessageGate ermöglicht den Zugang per aktivem FTP oder https/WebDAV. Als Übergabeformat dient eine SMTP Mitteilungsstruktur, die es dem Nutzer auf einfachem Weg ermöglicht, Mitteilungen zu erzeugen und zu verarbeiten. Sollte es sich bei den Daten um ein EDIFACT Interchange handeln, kann die zentrale EDI-Funktion von BusinessMail X.400 genutzt werden, die in der Lage ist, das Dokument auch ohne zusätzliche SMTP-Headerinformationen zu übertragen.

2 Vorschlag

EDI4RCom [Loh08] Um das EDI-Hub noch interessanter zu gestalten, wird ein Kommunikationsmodul entwickelt, welches als Gateway zwischen EDI-Hub und MessageGate eingesetzt werden kann. Dadurch sind EDI-Hub Nutzer in der Lage, innerhalb des Hubs mit den angebundenen Geschäftspartnern wie gewohnt zu kommunizieren. Zusätzlich können sie nun aber auch mit Geschäftspartnern in Verbindung treten, die nicht an das Hub angeschlossen sind, sondern über das X.400-Netzwerk kommunizieren. Von einer kleinen Insellösung wird das Hub damit zum Zugangspunkt für ein weltweites Kommunikationsnetzwerk ohne dabei seine Vorteile bezüglich der einfachen Bedienung und der geringen Kosten ein zu büßen. Da es auch vorstellbar ist, dass ein einzelnes Unternehmen eine einfache Verbindung zum X.400-Netzwerk sucht, ist es ebenfalls möglich das Kommunikationsmodul als Standalone-Lösung ohne EDI-Hub einzusetzen.

3 Details

EDI4RCom ist eine in Ruby-on-Rails[Han] entwickelte Open Source Lösung zum Geschäftsdatenaustausch unter Nutzung von Internetstandards. Es basiert EDI4R[Wer06], einer ebenfalls frei verfügbaren Bibliothek und Tool-Sammlung zum Umgang mit EDIFACT und EANCOM Dokumenten. Es steht als fertige Anwendung und als Ruby-Gem zur Integration in eine bestehende Anwendung zur Verfügung. Dadurch ist es möglich, auf Basis von EDI4R eine komplette EDI-Infrastruktur aufzubauen und das Kommunikationsmodul vollständig zu integrieren. Das Kommunikationsmodul ist so ausgelegt, dass es mit einfachen Mitteln als Gateway zwischen einem EDI-Hub und MessageGate fungieren kann. Für die Nutzer des Hubs soll nach einer einmaligen Konfiguration möglichst transparent sein, ob sie mit direkt angebundenen Partnern im Hub oder über das Gateway kommunizieren. Der Einsatz als Standalone-Lösung ist ebenfalls möglich. Die Unternehmenssoftware kann in diesem Fall direkt über eine Schnittstelle mit dem Modul kommunizieren. In jedem Fall wird ein MessageGate-Account benötigt, über den der Datenaustausch mit BusinessMail X.400 durchgeführt wird.

3.1 Kommunikation

Das Modul kommuniziert mit MessageGate über https mit WebDAV Erweiterung. Eine Verbindung über FTP kommt nicht in Frage, da hierfür zusätzlich ein VPN-Tunnel erforderlich wäre, was zusätzlichen Einrichtungsaufwand bedeuten würde. Mit der WebDAV Erweiterung bietet https zudem einen zu FTP vergleichbaren Leistungsumfang an. Allerdings mit dem Vorteil, dass im Gegensatz zu FTP nur eine TCP-Verbindung zum Server aufgebaut werden muss. Neben gewöhnlichen http/https Funktionen, wie GET und POST bietet WebDAV auch Funktionen zum Löschen, Verschieben und Anlegen ganzer Dateien oder Verzeichnisse auf dem Server. Um einen sicheren Zugang zum BusinessMail X.400 System zu gewährleisten wird hier neben der Authentifizierung des Servers durch ein Zertifikat auch ein Clientzertifikat benötigt. Das Kommunikationsmodul nutzt dafür die in Ruby bereits vorhandene Funktionalität zur Kommunikation über https und WebDAV, inklusive der Unterstützung von Clientzertifikaten. Außerdem kann es auch so konfiguriert werden, dass es mit MessageGate über einen Proxyserver kommuniziert. Eine Integration in ein bestehendes Firmennetzwerk soll

so erleichtert werden.

3.2 Verwaltung

Ein Webinterface ermöglicht die einfache Verwaltung der Anwendung. Neben der Ablaufsteuerung existiert auch eine Partnerverwaltung und eine Verwaltung für das Tracking des Mitteilungsverlaufs. MessageGate erfordert, abgesehen von dem Fall, dass die zentrale EDI-Funktion von BusinessMail X.400 eingesetzt werden soll, eine SMTP-Headerstruktur, um Mitteilungen zu verarbeiten. Es ist allerdings relativ unkomfortabel, jedes Dokument, das verschickt werden soll, per Hand um diese Informationen zu ergänzen. Darum werden in der Partnerverwaltung zu jedem Kommunikationspartner einmalig eingepflegt. Das Modul ergänzt dann die Mitteilungen um die erforderlichen Informationen. Bei eingehenden Mitteilungen werden diese entsprechend wieder entfernt, so dass der Nutzer lediglich mit den reinen Nutzdaten hantieren muss. Zu den erforderlichen Daten gehört neben einer eindeutigen ID, wie einer ILN, auch eine X.400-Adresse. Im Rahmen des Betriebs als Gateway für das EDI-Hub, kann das Interface in das Webfrontend des Hubs integriert werden. Falls das Webinterface nicht genutzt werden soll, kann das Modul auch über Konfigurationsdateien und entsprechende Einträge in der Datenbank manuell verwaltet werden.

3.3 Schnittstelle

Als lokale Schnittstelle zwischen Kommunikationsmodul und Nutzeranwendung dient eine Verzeichnisstruktur. Diese ist der Struktur des EDI-Hubs nachgebildet, um eine Integration zu erleichtern. Die Ordner werden allerdings aus Sicht des Kommunikationsmoduls mit anderen Bedeutungen benutzt. Es existieren folgende Unterverzeichnisse:

in Enthält Dateien, die von EDI4RCom verarbeitet werden sollen

out In dieses Verzeichnis werden eingehende Dateien geladen

stat Hier werden Dateien mit Statusätzen zu versendeten Dateien angelegt

err Verzeichnis zum Sammeln fehlerhafter Dateien

tmp Verzeichnis zur Aufnahme temporärer Dateien

Im Dateinamen werden eindeutige Sender-ID, Empfänger-ID sowie die Mitteilung-ID angegeben. Diese Konvention entspricht ebenfalls der des EDI-Hubs. So ist es ohne weitere Änderungen an den Dateinamen möglich, das Modul als Gateway für das EDI-Hub einzusetzen. MessageGate benutzt allerdings andere Vorgaben für Dateinamen. Daher muss innerhalb des Moduls eine Umsetzung durchgeführt werden.

3.4 Tracking und Tracing

Das Kommunikationsmodul protokolliert alle Verarbeitungsschritte, die es durchführt, in einer eigenen Datenbank. Außerdem fordert es von MessageGate in regelmäßigen Abständen Statusreports an und ergänzt seine eigenen Informationen durch die Informationen aus diesen Reports. Damit ist der Verlauf einer Mitteilung vom Öffnen durch das Kommunikationsmodul bis zur Zustellung beim Empfänger sowie von der Auslieferung einer Mitteilung an MessageGate bis zur endgültigen Auslieferung an der lokalen Dateischnittstelle festgehalten. Die gesammelten Informationen können als Statusdatei entsprechend der Struktur des EDI-Hubs im Verzeichnis *stat* bereitgestellt werden. Außerdem ist vorgesehen, die Informationen in die Tracking & Tracing Funktion des EDI-Hubs einzubinden. Dadurch ist ein Nutzer des Hubs über das integrierte Webinterface auch weiter vollständig über den Verlauf der Mitteilungen informiert.

3.5 Programmablauf

Um einen automatischen Betrieb der Anwendung zu ermöglichen, sind die Kernfunktionen des Moduls in einem einzelnen Daemon-Prozess implementiert, der die Überwachung von lokalem und MessageGate-Verzeichnis übernimmt, Mitteilungen verarbeitet und die Protokollierung durchführt. Treten Probleme in der Verarbeitung von Mitteilungen auf, besteht ein Verbindungsproblem zu MessageGate oder gibt es Fehler bei der Kommunikation mit der Datenbank, ist das Modul in der Lage, über verschiedene Wege Alarmierungen an die verantwortlichen Personen zu senden. Allerdings sollten im Wirkbetrieb zusätzliche Maßnahmen ergriffen werden, um die Anwendung selbst zu überwachen.

Das Programm prüft, wie bereits erwähnt, in regelmäßigen Abständen das lokale und das MessageGate- Verzeichnis auf Mitteilungen, die es verar-

beiten kann. Liegt im lokalen Verzeichnis eine Datei zur Verarbeitung vor, wird diese in das temporäre Verzeichnis kopiert. Dort werden die Informationen aus dem Dateinamen ausgelesen und die Datei geöffnet, um sie mit den SMTP-Headerinformationen zu erweitern. Können die Informationen nicht ausgelesen und ermittelt werden, zum Beispiel wenn kein entsprechender Partner angelegt wurde oder schlägt die Erweiterung der Datei fehl, wird die Datei in den Fehlerordner verschoben. Ist der Prozess erfolgreich, wird die Datei gemäß den Konventionen von MessageGate umbenannt und in das MessageGate-Verzeichnis transferiert. Anschließend wird die lokale Datei entfernt.

Liegt im MessageGate-Verzeichnis eine eingehende Mitteilung vor, wird diese in das lokale temporäre Verzeichnis kopiert. Dort werden die Adressinformationen entfernt und die Datei umbenannt, so dass Sender, Empfänger und eine ID im Dateinamen erkennbar sind. Die Aufzeichnung in der Datenbank geschieht, wie oben beschrieben, jeweils bei Ausführung der einzelnen Verarbeitungsschritte.

3.6 Voraussetzungen

Je nach Datenvolumen sollte ein gewöhnlicher Breitband-Internetzugang, wie ein vorhandener DSL-Anschluss genügen, um die Kommunikation zwischen dem Modul und MessageGate durchzuführen. Werden große Datenmengen übertragen, sollte aber sicher gestellt werden, dass dadurch nicht der weitere Betrieb beeinträchtigt wird. MessageGate kann derzeit mit einer Bandbreite von bis zu 34MBit angesprochen werden. Um das Modul zu betreiben, können verschiedene Betriebssysteme, wie Windows, Unix/Linux und Mac OS X eingesetzt werden. Grundvoraussetzung ist Ruby mit installiertem Rails als RubyGems-Paket. Für die Speicherung der Daten kommt eine Datenbank zum Einsatz. Ruby unterstützt neben MySQL und Oracle auch MS SQL Server oder PostgreSQL und viele Weitere. Für Rails wird ein WebServer wie Apache oder lighttpd benötigt.

Um eine leichte Anbindung an interne Unternehmensanwendungen zu ermöglichen, sollte das Modul auf einem Server innerhalb des Firmennetzwerkes eingerichtet werden. Wird es als Gateway für das EDI-Hub eingesetzt, kann es ebenfalls auf dem "managed root server" betrieben werden, auf dem auch das Hub arbeitet. Dieser erfüllt dann bereits alle weiteren Voraussetzungen für den Einsatz des Moduls, da auch das EDI-Hub alle

angegebenen Komponenten benötigt. Je nach dem Datenvolumen, welches über das Modul ausgetauscht wird, kann der Betrieb auf einem separaten System erforderlich werden.

3.7 Kosten

Da das Kommunikationsmodul verbreitete Internetstandards nutzt, um mit MessageGate zu kommunizieren und ein Breitband-Internetzugang in den meisten Unternehmen heute bereits vorhanden ist, fallen hier keine zusätzlichen Kosten an. Lediglich für den MessageGate-Zugang selbst entstehen zusätzliche Kosten. Das Preismodell ist aber derzeit noch nicht festgelegt, weshalb hier keine konkreten Angaben gemacht werden können. Um den Kostenvorteil des Hubs sicher zu stellen, teilen sich die Hub-Nutzer die anfallenden Fixkosten für den MessageGate-Zugang. Die Kosten für die Nachrichtenübertragung können durch das erweiterte Tracking & Tracing im EDI-Hub transparent und verursachergerecht aufgeteilt werden.

4 Zusammenfassung

EDI4RCom vervollständigt eine Open Source EDI-Infrastruktur basierend auf EDI4R durch ein Bindeglied zu MessageGate und ermöglicht so den Zugang zu BusinessMail X.400. Damit ist es nun erstmals möglich, mit frei verfügbarer, plattformunabhängiger Software an standardisierten, EDI-gestützten ECR-Prozessen teilzunehmen und gleichzeitig die Vorteile eines weltweiten Netzwerks zu nutzen.

Weitere Vorteile:

- Nutzung von Internetstandards und Sicherheit durch X.400
- Kosten- und Aufwandsreduzierung in Einführungsphase und Betrieb
- Zustandstracking dank Unterstützung von X.400-Reports
- Anpassbarkeit an eigene Ansprüche

Literatur

- [Han] David Heinemeier Hansson. Ruby on rails. Website. Projektseite: <http://www.rubyonrails.org/>.
- [Loh08] Martin Lohrmann. Edi4rcom. Website, 2008. Projektseite: <http://www.informatik.fh-wiesbaden.de/~werntges/proj/edi4rcom01.html>.
- [TS] T-Systems. Businessmail x.400. Website. Produktseite: <http://www.mittelstand.t-systems.de/msp/cms/content/MSP/de/business-mail-x400>.
- [Wer06] Heinz W Werntges. Edi for ruby. Website, 2006. Projektseite: <http://edi4r.rubyforge.org/>.
- [Wer07] Heinz W Werntges. Edi-hub. Website, 2007. Projektseite: http://www.informatik.fh-wiesbaden.de/~werntges/proj/edi_hub01.html.