Bachelorarbeit

Umsetzung von Logging-Richtlinien und Einrichtung eines zentralisierten Logging-Servers für das CFT Portale der Kassenärztlichen Vereinigung Westfalen-Lippe

Kevin Bollich
geboren am 19.04.1997
Matr.-Nr.: 7102160

An der Fachhochschule Dortmund im Fachbereich Informatik erstellte Bachelorarbeit
im Studiengang Software- und Systemtechnik Dual - Vertiefungsrichtung Softwaretechnik

zur Erlangung des akademischen Grades
Bachelor of Science
B. Sc.

Betreuung durch:
Prof. Dr. Martin Hirsch

4. November 2020

Inhaltsverzeichnis

Abbildungsverzeichnis

1. Einführung

1.1. Motivation

Das CFT Portale der Kassenärztlichen Vereinigung Westfalen-Lippe (KVWL) verwaltet eine hohe Anzahl an Applikationen, bei denen regelmäßig neue Funktionen hinzukommen. Bei der stetigen Weiterentwicklung können während der Laufzeit Fehler auftreten, deren Herkunft nicht immer eindeutig ist. Damit die Herkunft solcher Fehler erkannt werden kann, sollten bestimmte Laufzeitinformationen geloggt werden. Da derzeit keine klare Struktur im Logging erkennbar ist, ist das Bugtracking im CFT Portale sehr zeitaufwendig. Der Grund dafür liegt hauptsächlich an der redundanten Serververteilung und dem unstrukturierten Logging.

1.2. Problemstellung

Im CFT Portale müssen die Entwickler regelmäßig die Ursachen von aufgetretenen Fehlern analysieren. Dabei sieht der Prozess folgendermaßen aus:
Jede Anwendung ist auf zwei redundanten Servern installiert und schreibt auf dem jeweiligen Server ihre Logs. Damit die Entwickler herausfinden können, wo das entsprechende Log geschrieben wurde, muss auf beiden Servern manuell nach dem Fehler gesucht werden. Da ein Fehler nicht immer sofort nach Auftreten gemeldet wird und die Software trotz des Fehlers weiter läuft, steigt die Menge an geschriebenen Logs. Den Fehler in den Logdateien zu finden, kann durch die fehlende Möglichkeit des Filterns sehr zeitaufwendig werden.
Eine weitere Herausforderung bei der Fehlersuche liegt in den unstrukturierten Informationen in den Logdateien.

Daraus leiten sich folgende Forschungsfragen ab:

- Mit welchem Log-Management-Tool ist ein an die Probleme des CFT Portale angepasstes zentralisiertes Logging möglich?
- Wie kann ein zentralisierter Logging-Server eingerichtet werden?
- Können die Logging-Richtlinien aus der Projektarbeit in der Praxis umgesetzt werden?

1.3. Zielsetzung

Ziel dieser Bachelorarbeit ist es, die in der Projektarbeit definierten Logging-Richtlinien anhand der Software „Vierteljahreserklärung“ durchzuführen. Außerdem soll eine Evaluierung von Log-Management Tools erfolgen, damit herausgefunden werden kann, ob der in der Projektarbeit erwähnte Elastic-Stack die beste Lösung für das CFT Portale ist, um ein zentralisiertes Logging einzurichten. Wenn die Entscheidung über das Log-Management Tool gefunden wurde, soll ein zentralisiertes Logging mit dem Log-Management Tool umgesetzt werden.

1.4. Vorgehensweise

Zu Beginn der Bachelorarbeit erfolgt eine Evaluation von Log-Management-Tools. Mithilfe der Evaluation soll ein passendes Tool identifiziert werden, dass eine effiziente zentralisierte Lösung für das CFT Portale ermöglicht. Bevor dies geschieht, müssen noch die Anforderungen an das Tool aufgestellt werden. Dies geschieht in Absprache mit dem CFT Portale. Nachdem ein Tool identifiziert wurde, soll ein zentralisierter Logging-Server eingerichtet werden. Dieser soll in Zukunft die erstellten Logs sammeln, anzeigen und analysieren. Anschließend sollen in der Applikationen „Vierteljahreserklärung“ alle in der Projektarbeit definierten Richtlinien umgesetzt werden. Zum Schluss wird ein Fazit zum Verlauf der Bachelorarbeit gezogen. Dabei werden die Ergebnisse der Arbeit noch einmal vorgestellt und bewertet.

Vorläufige Gliederung:

- Einführung

- Evaluation von Log-Management Tools
- Einrichten eines zentralisierten Logging-Server
- Umsetzen der Richtlinien
- Fazit

2. Evaluation von Log-Management Tools

In der vorherigen Projektarbeit wurden für das CFT Portale Richtlinien definiert, die in dieser Bachelorarbeit praktisch umgesetzt werden sollen. Eine dieser Richtlinien war die Nutzung von einem zentralisierten Logging-Server mithilfe des Elastic Stack. Jedoch wurden in der Projektarbeit keine weiteren Tools herangezogen, um zu prüfen, ob der Elastic Stack die beste Alternative ist.
In diesem Kapitel werden unterschiedliche Tools, die für das Log-Management genutzt werden können, evaluiert. Das Ziel dieser Evaluation ist zu prüfen, ob es eine bessere Alternative für eine zentralisierte Logging Lösung gibt, als den Elastic Stack. Dafür werden Tools evaluiert, die den kompletten Elastic Stack ersetzen können, aber auch Tools die einzelne Komponenten austauschen können.
Das CFT Portale wäre in der Lage weitere Kosten für ein Tool auf sich zu nehmen, sollte es dem Team die Arbeit erleichtern können. Daher werden Open-Source und Lizenzpflichtige Tools in dieser Evaluation betrachtet. Sollten jedoch zwei Tools gleichermaßen die Anforderungen erfüllen und eines der Tools Open-Source sein, dann wird sich für das Open-Source Tool entschieden, um kosten zu sparen.
Damit eine Evaluation erfolgen kann, müssen Anforderungen aufgestellt werden. Die Anforderungen sollen dabei helfen eine Entscheidung bezüglich der Tools treffen zu können. Denn Tools die diese Anforderungen nicht erfüllen können, werden nicht weiter betrachtet. Im nächsten Abschnitt werden die Anforderungen definiert.

2.1. Anforderungen an die Log-Management Tools

In diesem Abschnitt werden Anforderungen für die zentralisierten Logging Tools definiert. Die Anforderungen helfen bei der Entscheidung, ein passendes Tool für

das CFT Portale auszuwählen. Daher wurden in Absprache mit dem Team einige Anforderungen definiert, die das zukünftige Tool haben sollte. Für das CFT Portale spielt Wartung eine wichtige Rolle, daher werden einige Anforderungen sich auf den Wartungsaufwand beziehen.

Da das KV-Netz Sicherheitstechnisch stark abgeschirmt ist, kommt eine Cloud Lösung nicht in Frage. Das bedeutet, dass das Tool eine Selbstorganisierte Lösung bieten muss.
Durch die Menge an Anwendungen die das CFT Portale betreuen muss, ist es wichtig, dass das Tool die einzelnen Logs entweder von den unterschiedlichen Maschinen selbst einsammeln kann oder ein Senden von den Anwendungen heraus möglich ist. Das installieren weiterer Log-Agenten die für das schicken der Logs zuständig wäre sollte vermieden werden. Da sonst weiterer Wartungsaufwand entstehen würde.
Da das CFT Portale sich um keine Datenbanken kümmert ist der aktuelle wissensstand des Teams eher allgemein vorhanden. Denn das Team übernimmt in der Regel die Entwicklung von Provider-hosed Apps im SharePoint Umfeld. Damit das Team also erfolgreich mit den Datenbanken arbeiten kann, müssen Schulungen absolviert werden. Daher ist eine wichtige Anforderung die Speicherung der Logs. Das ausgewählte Tool muss eine Möglichkeit anbieten die Logs zu speichern.
Ein wichtiger Punkt ist die Analyse und Anzeige von Logs. Damit ist eine Oberfläche gemeint, die intuitiv benutzt werden kann, um die Logs anzusehen und zu Analysieren. Jedoch sollte in dem Tool auch die Möglichkeit bestehen, Logs zu Filtern und zu durchsuchen.
Im CFT Portale sind Linux- und Windows Server im Betrieb. Jedoch möchte das Team das Tool gerne auf einer Linux Maschine installieren. Da dort das updaten von neuen Versionen einfacher funktioniert und der wissensstand des Teams da komplett gegeben ist.

Das waren die Vorgaben die Vom CFT Portale für ein Tool definiert wurden. Hier nochmal eine kleine Aufzählung der einzelnen Anforderungen:

- Selbstorganisierte Lösung (Kein CLoud)
- Einsammeln von Logs aus unterschiedlichen Anwendungen

- Speicherung von Logs ohne externe Datenbank
- Anzeige und Analyse von Logs
- Filtern und Dursuchen von Logs
- Installation auf Linux Server

2.2. Log-Management Tools

In der Projektarbeit wurde der Elastic Stack in seiner Funktionsweise und dessen Möglichkeiten schon ausreichend vorgestellt. Daher wird in diesem Kapitel hauptsächlich auf die neu zu evaluieren Tools eingegangen. Alle Tools werden hier einmal vorgestellt mit all ihren Vor- und Nachteilen.

2.2.1. Graylog

Graylog ist ein Open-Source Log-Management Tool. Dessen Motto ist:

> „less cost, more performance“[Gra]

Das Tool setzt auf Performance. Die Funktionalitäten des Tools beziehen sich auf das Sammeln, verbessern, Speichern und die Analyse der Logs. In Graylog kann man eigene Dashboards erstellen und individuell anpassen. Dabei wird mithilfe von Suchabfragen das Dashboard definiert. Damit nicht jeder Mitarbeiten sich mit den Suchabfragen beschäftigen muss, können die Dashboards geteilt werden. Graylog bietet jedoch auch vordefinierte Dashboards an, die genutzt werden können.
Mithilfe von Graylog können unmengen an Logs gespeichert werden. Daher ist die Suche in Großen Datenmengen sehr wichtig. In Graylog werden die Logs beim Speichern sofort indiziert, um eine schnelle Suche zu ermöglichen. Die Daten werden beim Speichern geprüft ob sie eine gute struktur aufweisen. Wenn die Struktur nicht gut ist, wird sie verbessert. Die Architektur von Graylog ermöglicht eine multi-threaded Suche. Jede Suche nutzt dabei mehrere Prozessoren um möglichst effizient zu sein. Die Suche in Graylog ist dabei einfach aufgebaut. Einfache Boolean Operationen werden für die Suche genutzt. Die dazu benötigten Felder werden durch einfaches

klicken ausgewählt. Damit muss keine neue Suchsprache erlernt werden und kann auch von nicht Fachpersonal genutzt werden. [Gra]

Graylog kann als Enterprise Variante gekauft werden. Bei der Enterprise Variant wird Support angeboten, sowie Features die in der Open Source Variante nicht vorhanden sind. Zu dem Support gehört Hilfe zu allen Graylog bezogenen Fragen, jedoch bietet Graylog zusätzlich Support für weitere Komponenten an, die zusätzlich installiert werden müssen. Zu den zusätzlich zu installierenden Software Komponenten gehören: Elasticsearch, MongoDB und Oracle Java SE 8 (oder OpenJDK 8).

Graylog wirbt mit folgenden Funktionen: Massive Skalierbarkeit, Performance, Anzeige und Analyse von Log,

2.2.2. Loggly

2.2.3. Splunk

2.2.4. LogDNA

2.2.5. Fluentd

Literatur

[Gra] Graylog. *Industry Leading Log Management | Graylog*. URL: https://www.graylog.org/ (besucht am 02. 11. 2020).

A. Anhang

	OPEN SOURCE Contact sales
Extended log collection using Sidecar	✓
Scalable log collection	✓
Log enrichment data	✓
Simple UI for administration	✓
Graphical log analysis	✓
Content Packs	✓
Alerts & Triggers	✓
REST API	✓
Free marketplace of extensions	✓
LDAP integration	✓
Correlation Engine	
Scheduled Reports	
Data Forwarder	
Offline log Archiving	
User Audit Logs	
Search Parameters	
Technical Support	
Search Workflows	

A.1. Eideststattliche Erklärung

Eideststattliche Erklärung
Ich versichere an Eides statt, dass ich die vorliegende Arbeit selbständig angefertigt und mich keiner fremden Hilfe bedient sowie keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Alle Stellen, die wörtlich oder sinngemäß veröffentlichten oder nicht veröffentlichten Schriften und anderen Quellen entnommen sind, habe ich als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Dortmund, den 31.08.2020

Kevin Bollich

Erklärung
Mir ist bekannt, dass nach § 156 StGB bzw. § 163 StGB eine falsche Versicherung an Eides Statt bzw. eine fahrlässige falsche Versicherung an Eides Statt mit Freiheitsstrafe bis zu drei Jahren bzw. bis zu einem Jahr oder mit Geldstrafe bestraft werden kann.

Dortmund, den 31.08.2020

Kevin Bollich