
Dokumentation zur betrieblichen Abschlussarbeit

# Evaluation, Installation, Konfiguration und Inbetriebnahme eines zentralisierten Logging-Systems

Durchgeführt von:

**Kai Müller**

27.03.1992, Barßel

Ausbildungsberuf:

**Fachinformatiker für Systemintegration**

Ausbildungsbetrieb:

**CEWE Stiftung & Co. KGaA**

Projektbetreuer:

**Klaus Tornow-Frerichs**

11. Mai 2022

# Inhalt

# Tabellenverzeichnis

# Abbildungsverzeichnis

# 1 Vorwort

Mein Name ist Kai Müller und ich absolviere momentan meine Ausbildung zum Fachinformatiker für Systemintegration bei der CEWE Stiftung & Co. KGaA (im Folgenden CEWE). Das Projekt wurde im Rahmen meiner betrieblichen Ausbildung für CEWE geplant, durchgeführt und dokumentiert. Aus Gründen des Datenschutzes wurden in dieser Dokumentation sämtliche IP-Adressen, Hostnamen und Passwörter unkenntlich gemacht oder durch Platzhalter ersetzt.

# 2 Definitionsphase

Im nachfolgenden Kapitel werden das Projektumfeld, die Beschreibung des Projektes sowie dessen Ziel beschrieben. Zuerst wird auf das Unternehmen eingegangen, in dem das Projekt umgesetzt wird und darauffolgend der Inhalt und das Ziel des Projektes erörtert.

## 2.1 Projektumfeld

Das Projekt wird in der CEWE Stiftung & Co. KGaA mit Hauptsitz in Oldenburg durchgeführt. Die CEWE Unternehmensgruppe hat sich als hochqualitativer Dienstleister im Bereich des Foto- und Online-Druckservices etabliert. In 14 Produktionsstandorten und zehn Vertriebsniederlassungen werden etwa 4.000 Mitarbeiter in ganz Europa beschäftigt. Im Jahr 2021 wurden rund 2,18 Mrd. Fotos und 5.65 Mio. CEWE Fotobücher ausgeliefert. Andere Produkte sind zum Beispiel die CEWE Cards, die CEWE Wandbilder, die CEWE Kalender oder die CEWE Fotogeschenke. So konnte die Unternehmensgruppe im Geschäftsjahr 2021 einen Umsatz von 692,8 Mio. Euro erzielen.[1]

Das Projekt wird in der Abteilung Produktions-IT durchgeführt. Der Bereich der Produktions-IT bereitet die Rohdaten der Kundenbestellungen für die Produktion vor. Hier werden zum Beispiel mehrere Aufträge zusammengefasst, die Position der Bilder für die unterschiedlichen Druckmaschinen angepasst oder die Label für den Versand der Produkte erzeugt. Die Abteilung sorgt für den reibungslosen Ablauf in der Produktion und stellt die Hintergrundprozesse rund um die Produktion bereit.

## 2.2 Projektbeschreibung

Auf fast allen Systemen in der Produktion werden Prozesse ausgeführt, die ihre diversen Tätigkeiten in Log-Dateien schreiben. Momentan muss sich ein Administrator händisch am jeweiligen System anmelden, damit er die Log-Dateien abrufen und durchsuchen kann. Diese Anmeldung kann entweder vor Ort am jeweiligen System erfolgen oder per Remote Zugriff, z.B. via Virtual Network Computing (VNC) oder Secure Shell (SSH). Der Workflow der Fehlersuche soll durch eine Softwarelösung zentralisiert und insgesamt optimiert werden, um den Zeitaufwand zu reduzieren.

[1] CEWE Geschäftsbericht 2021, URL: https://ir.cewe.de/download/companies/cewe/Annual%20Reports/DE0005403901-JA-2021-PN-EQ-D-00.pdf (abgerufen am 02.05.2022)

## 2.3 Projektziel

Das Ziel des Projektes ist es den Zugang auf die Log-Dateien zu zentralisieren und den Arbeitsaufwand reduzieren. Dies soll durch eine zentralisierte Softwarelösung erreicht werden. Die Log-Dateien sollen automatisiert und in Echtzeit von den Clients abgerufen und an das zentrale Logging-System verschickt werden. Hier sollen die Log-Dateien über eine grafische Oberfläche angezeigt und mithilfe einer Suchmaske durchsucht werden können. Zu den eigentlichen Log-Zeilen sollen ebenfalls Daten, wie der Hostname des Clients, IP-Adresse und Systemzeit übermittelt werden, um die gezielte Suche nach Systemdaten zu ermöglichen.

# 3 Planungsphase

In dieser Projektphase wird der Ist-Zustand untersucht und ein Soll-Konzept durch Gespräche mit den Mitarbeitern und Administratoren erstellt. Das Projekt wird in mehrere Projektphasen unterteilt und zeitlich gegliedert. Es wird ein Softwarevergleich und eine Nutzwertanalyse als Entscheidungshilfe erstellt und zusätzlich ein Angebot für einen physischen Servern eingeholt und den Kosten des Betriebs einer virtuellen Maschine gegenübergestellt.

## 3.1 Konzepterstellung

Im Folgenden wird eine Ist-Analyse vorgenommen und ein Soll-Konzept erstellt, um das Vorgehen beim Projekt zu vereinfachen und zu strukturieren.

### 3.1.1 Ist-Analyse

Auf den IT-Systemen der Produktion am Standort Oldenburg werden diverse Programme und Tools, die Bilddaten bearbeiten, ändern, verbessern oder erweitern, bereitgestellt. Die meisten dieser Tools werden auf VM‘s, virtuellen Maschinen, ausgeführt. Als Beispiel werden Bilder mit niedrigen Auflösungen automatisch verbessert, rote Augen korrigiert und Dateiformate geändert, um die Dateien für verschiedene Maschinen vorzubereiten. Diese Systeme dokumentieren ihre Arbeitsschritte in Log-Dateien. Gibt es Probleme bei den Arbeitsschritten oder treten Fehler auf, werden diese ebenfalls in die lokalen Log-Dateien geschrieben. Dies ist wichtig, um später zu gewährleisten, dass Administratoren die Fehler nachverfolgen und nachvollziehen können.

Momentan erfolgt die Nachverfolgung dadurch, dass ein Administrator entweder von Mitarbeitern in den Produktionsbereichen oder durch das automatische Monitoring der IT-Systeme benachrichtigt wird, dass es ein Problem gibt oder etwas nicht plangemäß funktioniert, wie es sollte. Der Administrator meldet sich dann per Remotezugriff, je nachdem welches Betriebssystem auf dem Problemsystem verwendet wird, entweder über VNC (bei Windowssystemen) oder SSH (bei Linuxsystemen) auf den Systemen an. Dort navigiert der Administrator dann in die entsprechenden Unterordner, die die Log-Dateien enthalten. Unter Windows ist dies meistens $C{:}\backslash Programme\backslash < Software - Name > \backslash Logs$ und unter Linux ist dies meist $/home/**** **/< Software >/logs/$ . Der Administrator ruft dann die für den Problemfall relevante Log-Datei auf. Die Log-Dateien werden dann mit einem Text-Editor geöffnet und nach auffälligen Zeilen durchsucht. Je nach Erfahrungsstand des Administrators wird hier entweder nach system-typischen Fehlermeldungen gesucht oder erst einmal nach Zeilen, die „Error“ oder

„Warning“ enthalten und ungefähr in das Zeitfenster des Fehlerbildes passen. Leider sind die Log-Dateien oftmals sehr umfangreich (>50000 Zeilen für einen einzigen Werktag), was die Fehlersuche sehr zeitaufwendig und auch fehleranfällig macht. Übersieht der Administrator den Fehler muss er später möglicherweise noch einmal dieselbe Log-Datei aufrufen und suchen, falls das Problem nicht anderweitig behoben werden konnte.

### 3.1.2 Soll-Konzept

Der momentane Arbeitsablauf bei der Fehlersuche in Log-Dateien ist umständlich, zeitintensiv und fehleranfällig. Daher muss dieser Prozess modernisiert und zentralisiert werden. Dafür soll ein System konzipiert werden, dass das zentrale Sammeln und Durchsuchen der Log-Dateien ermöglicht. Hierfür muss einerseits die Software als auch die benötigte Hardware unter kaufmännischen und technischen Gesichtspunkten betrachtet werden. Das neue System soll on-Premise, also im betriebsinternen Rechenzentrum, zum Einsatz kommen. Dabei soll das System entweder auf einem physischen Server oder einer virtuellen Maschine zum betrieben werden. Der entsprechenden Abteilung ist es zusätzlich wichtig, dass eine Open-Source Software zum Einsatz kommt.

Das System soll eine statische IP-Adresse und einen DNS-Eintrag erhalten. Wichtig ist, dass das System nur aus dem internen Netzwerk des Unternehmens erreichbar ist, es soll nicht über das Internet erreichbar sein. Der neue Arbeitsablauf soll dem Nutzer ermöglichen sich an das zentrale Logging-System anzumelden und dort nach gewissen Keywords, also Schlüsselwörtern, zu suchen. Diese könnten zum einen Hostnamen und IP-Adressen oder auch die Wörter „Error“, „Warning“ oder „Alert“ sein.

Auch eine Gruppierung der Maschinen wäre wünschenswert, damit Log-Dateien schon vorab sortiert werden und der Administrator Suchbefehle nicht auf alle vorhandenen Log-Dateien anwenden muss. Die Gruppierung soll sowohl nach Maschinentyp als auch Betriebssystem möglich sein.

Der Umfang der Systeme, von denen Log-Dateien gesammelt werden sollen, soll sich auf den Bereich der Produktion des Standorts Oldenburg beschränken. Im Umfang dieses Projekts sollen vorerst die Log-Dateien von ein bis zwei Systemen integriert werden. Wenn das Projekt in seiner Form, wie es hier geplant wird, gut genutzt und akzeptiert wird, sollen langfristig die Log-Dateien von etwa 200 Maschinen gesammelt werden. Neben den Log-Dateien der hausinternen Software, die auf den diversen Systemen läuft, soll es möglich sein Log-Dateien von Systemdiensten, wie z.B. Syslog, zu sammeln. Administratoren müssen Berechtigungen erhalten, die es Ihnen ermöglicht Log-Dateien neuer Systeme in das zentrale Logging-System zu integrieren, Suchfilter anzupassen oder Gruppierungen von Systemen in der Suchmaschine zu erstellen.

## 3.2 Ablaufplan

Das Projekt ist in unterschiedliche Phasen eingeteilt. Die Auflistung der Phasen findet sich in der Tabelle 1.

| Schritt | Zeit (geplant) | Zeit (tatsächlich) |
|---|---|---|
| **Planungsphase** | **11** | **8** |
| **Durchführung** | **16** | **20** |
| **Abschlussphase** | **8** | **7** |
| **Gesamt** | **35** | **35** |

*Tabelle 1 Zusammenfassung der Projektphasen*

## 3.3 Vergleich der Software

Es gibt verschiedene Software, die die Aufgaben eines zentralen Logging-Systems erfüllen können. Während meiner Ausbildung bin ich sowohl mit Graylog als auch mit Elastic in Kontakt gekommen. Da die gewählte Software Open-Source sein muss kamen für dieses Projekt diese beiden Softwarelösungen in Frage und werden im Folgenden vergleichen.

### 3.3.1 Graylog

Graylog wird von dem nordamerikanischen Unternehmen Graylog Inc. mit Sitz in Houston, Texas entwickelt. Das Unternehmen wurde 2009 gegründet und beschäftigt zwischen 50 und 100 Mitarbeiter.[2] Graylog besteht aus einer einzigen Softwarekomponente, die sich mit kostenlosen und kostenpflichtigen Plugins erweitern lässt. Graylog bietet gegenüber dem ELK-Stack ein integriertes und kostenfreies Alarmierungssystem. Sollten zum Beispiel gewisse Imports von Log-Dateien nicht mehr funktionieren, kann Graylog den Administrator darüber informieren. Beim ELK-Stack ist dies ein kostenpflichtiges Feature. Dafür schneidet der ELK-Stack bei der Log-Analyse und Aufbereitung der Logs sowieso der Suchgeschwindigkeit deutlich besser ab. Graylog verfügt ebenso nur über eine sehr grundlegende Visualisierungskomponente. Administratoren, denen Visualisierung der Daten wichtig ist, müssen daher meist noch auf eine Lösung, wie Grafana oder Kibana des ELK-Stacks, setzen. Dafür ist der Administrationsaufwand bei Greylog geringer und der Einstieg in das System ist einfacher als beim ELK-Stack der Fall ist.

### 3.3.2 ELK-Stack

ELK-Stack bezeichnet den Software-Stack der Firma Elastic, der die Open-Source Komponenten Elasticsearch, Logstash und Kibana umfasst. Hierbei ist Elasticsearch die Komponente, die die Log-Dateien speichert und das Durchsuchen der Log-Dateien ermöglicht. Logstash dient zur Aggregation, Verarbeitung und Anpassung der Log-Dateien, bevor sie an Elasticsearch verschickt werden. Kibana bezeichnet die Visualisierungskomponente des Systems.

Bis 2010 agierten die Teams hinter den drei Projekten allein, schlossen sich aber 2010 zu einem Projekt zusammen, da die drei Komponenten gut harmonierten und aufeinander aufgebaut werden konnten, um den heute sehr bekannten ELK-Stack zu schaffen[3]. Der ELK-Stack überzeugt vor allem durch seine Flexibilität und den Funktionsumfang. Nicht nur können Daten zentral gesammelt und durchsucht werden, sondern auch die Visualisierung ist von Anfang an gegeben. Außerdem ist die Erstellung von Suchfiltern in wenigen Klicks möglich und die

[2] Graylog – About us. URL: https://www.graylog.org/about (abgerufen am 01.05.2022)
[3] History of Elasticsearch. URL: https://www.elastic.co/de/about/history-of-elasticsearch (abgerufen am 03.05.2022)

Geschwindigkeit der Suche war unter den getesteten Produkten beim ELK-Stack mit Abstand am höchsten.

Der ELK-Stack verfügt mit Logstash über eine eigene, separate Schnittstelle für die Verarbeitung, Analyse und Anreicherung von Log-Dateien. Da dies für das Projektumfeld sehr wichtig ist und langfristig viele verschiedene Systeme geloggt werden sollen, habe ich mich für den Einsatz des ELK-Stacks entschieden. Ausschlaggebend war für die Wahl ebenfalls die Nutzwertanalyse (vgl. Tabelle 2).

Für die Nutzwertanalyse wurden sechs für das Projekt wichtige Kriterien ausgewählt und mit Gewichtungen versehen. Umso wichtiger das Kriterium für das Gesamtprojekt ist, desto höher die Prozentzahl. Daraufhin wurden für jedes Kriterium für beide Produkte Punkte verteilt. Die Punkteskala reicht von einem Punkt bis zu fünf Punkten. Eine höhere Punktzahl repräsentiert eine besser geeignete Lösung für das Projekt. Die vergebenen Punkte werden anschließend mit der Gewichtung multipliziert und im Anschluss addiert. Die maximal erreichbare Punktzahl beträgt fünf (alle Kriterien wurden mit fünf Punkten bewertet) und die minimal erreichbare Punktzahl beträgt eins (alle Kriterien wurden mit einem Punkt bewertet). Das Produkt mit der höchsten Punktzahl repräsentiert die für das Projekt am besten geeignete Lösung.

| | | **Graylog** | | **ELK-Stack** | |
|---|---|---|---|---|---|
| **Eigenschaft** | **Gewichtung** | Pkt. | Gew. | Pkt. | Gew. |
| Log-Parsing | 35% | 3 | 1,05 | 5 | 1,75 |
| Visualisierung | 10% | 1 | 0,10 | 5 | 0,50 |
| Einrichten von Suchmustern | 10% | 2 | 0,20 | 4 | 0,40 |
| Dokumentation, Anleitung | 30% | 3 | 0,90 | 5 | 1,50 |
| Auswahl an Log-Shippern | 10% | 2 | 0,20 | 3 | 0,30 |
| Verfügbarkeit von Plugins | 5% | 3 | 0,15 | 4 | 0,20 |
| | **Ergebnis:** | 14 | 2,60 | 26 | **4,65** |

*Tabelle 2 Nutzwertanalyse*

## 3.4 Angebote der benötigten Hardware

Bei der Projektrealisierung kommen zwei Umsetzungsarten in Frage: Entweder wird der ELK-Stack als On-Premise Lösung auf dem Virtualisierungscluster der Produktions-IT umgesetzt oder es wird ein physischer Server gekauft und auf diesem der ELK-Stack installiert. Ein Vergleich der Kosten befindet sich unter diesem Absatz in Kurzform (vgl. Tabelle 3). Aufgrund der großen Menge an Log-Dateien, die das System potenziell gleichzeitig verarbeiten soll und der Tatsache, dass alle drei Komponenten des ELK-Stacks auf demselben Server betrieben werden sollen, habe ich mich für einen Server mit 2x32GB 3200MHz ECC Arbeitsspeicher entschieden. Da meine Recherche gezeigt hat, dass Arbeitsspeicher und Datenträgerspeicher eher limitierende Faktoren für die Geschwindigkeit des Systems sind als CPU-Rechenleistung habe ich mich, um Kosten zu sparen, für einen 6-Kern Prozessor der Intel Xeon Reihe entschieden. Um sowohl genug Platz als auch eine schnelle Verarbeitung der Log-Dateien zu gewährleisten, wird das System mit zwei 1,92TB SATAIII Samsung SSD's ausgestattet. Auch diese erhöhen den Preis des physischen Servers sehr, sind für die performante Nutzung des Systems aber unabdingbar. Der physische Server wurde beabsichtigt mit höheren

Hardwarekapazitäten kalkuliert, da Aufrüstmöglichkeiten nach Anschaffung nicht immer schnell umsetzbar sind.

| Art der Lösung | Anschaffungskosten | Lfd. Kosten pro Jahr | Gesamtpreis über Nutzungsdauer |
|---|---|---|---|
| physischer Server | 2.842,17€ | 1.388,57€ | 5.619,31€ |
| virtuelle Maschine | - | 339,45€ | 678,90€ |

*Tabelle 3 Kostenvergleich*

Nach Rücksprache mit dem Auftraggeber, dem Abteilungsleiter der Produktions-IT, wurde sich vorerst auf eine Nutzungsdauer von zwei Jahren geeinigt. Der Gesamtpreis über die Nutzungsdauer setzt sich zusammen aus Anschaffungskosten und den laufenden Kosten über zwei Jahren. Nach Ablauf der zwei Jahre möchte der Kunde eine erneute Analyse durchführen, um zu prüfen, wie das Projekt im Betrieb aufgenommen wurde und ob es zu einer tatsächlichen Reduzierung des Arbeitsaufwandes geführt hat. Da die Kosten eines physischen Servers die Kosten für den Betrieb einer VM übersteigen, wird das Projekt auf einer VM durchgeführt. Die genaue Kostenberechnung für den Betrieb der VM wird im Punkt sechs der Dokumentation detailliert beschrieben.

### 3.4.1 Erstellen der benötigten virtuellen Maschine

Das unter Abschnitt 3.4 erwähnte Cluster verfügt über folgende Hardware pro Host (6 Hosts pro Cluster):

| | |
|---|---|
| Prozessor | 64x Intel® Xeon® Gold 6242 CPU@2.8GHZ |
| Arbeitsspeicher | 1TB DDR4 ECC |
| Festplattenspeicher | 34 TiB Blockstorage über SCSI (für gesamtes Cluster) |
| Netzwerkanbindung | 2x 1Gbit Ethernet & 2x 10Gbit Fiber Channel |

*Tabelle 4 Hardware des Proxmox Clusters*

Standardmäßig werden virtuelle Maschinen auf dem Cluster mit 2 Prozessorkernen und 4GiB Arbeitsspeicher erstellt. Da sowohl Elasticsearch als auch Kibana und Logstash auf dieser virtuellen Maschine ausgeführt werden sollen, möchte ich die Maschine hardwareseitig eher stärker aufstellen. Daher erstelle ich eine Maschine mit 16 Prozessorkernen und 32 GiB Arbeitsspeicher. Um ausreichend Speicherplatz für die Log-Dateien zu bieten, wird zudem eine Festplatte mit 1 TiB Speicherplatz angelegt. Da das Nutzen einer virtuellen Maschine Vorzüge, wie Hardware-Virtualisierung, bietet wurde die virtuelle Maschine mit geringeren Kapazitäten gewählt, als der physische Server bietet. Sollten mehr Ressourcen benötigt werden, können diese bei der virtuellen Maschine innerhalb weniger Minuten zugewiesen werden.

## 3.5 Datenschutz

Im Projektumfeld ist es wahrscheinlich, dass Administratoren bei der Erfassung von Log-Dateien in Kontakt mit persönlichen Daten von Kunden kommen. Dies können zum Beispiel Log-Dateien von Versandsystemen sein, die Adressen von Kunden enthalten. Da auch diese Dateien für die Fehlerverfolgung relevant sind, sollen auch diese Daten ebenfalls an das Logging-System übertragen werden. Dies bedeutet allerdings, dass auch Nutzer des Systems, die keine Administratoren sind, Zugriff auf sensible Daten erhalten. Daher ist es wichtig, das Projekt auch aus einem Datenschutzaspekt zu betrachten, denn für CEWE hat der verantwortungsvolle Umgang mit Kundendaten eine hohe Priorität. Daher wurde im Verlauf des

Projektes beschlossen, dass Mitarbeiter, die Zugriff auf das Logging-System erhalten, eine IT-Sicherheitsrichtlinie unterschreiben, die auch das Thema Datenschutz abgedeckt. Damit soll gewährleistet werden, dass sich Nutzer des Systems über den richtigen Umgang mit kundenbezogenen Daten bewusst sind. Die relevante Passage der Richtlinie ist wie folgt:

„1.2 Datenschutz und Datensicherheit: ein starkes Team!

Der Datenschutz beschäftigt sich mit dem Schutz von personenbezogenen Daten. Nach der gesetzlichen Definition (§ 3 Abs. 1 Bundesdatenschutzgesetz, BDSG) sind personenbezogene Daten "Einzelangaben über persönliche oder sachliche Verhältnisse einer bestimmten oder bestimmbaren natürlichen Person (Betroffener)". Damit sind alle Informationen umfasst, die über eine Person etwas aussagen und einer Person zugeordnet werden können. Hierzu gehören z.B. Name, Adressdaten, Kontonummer, religiöse Zugehörigkeit etc. aber auch Daten wie IP-Adressen, Bilddaten (...mit z.B. Personen auf dem Bild) oder Trackingdaten. Hierzu gehören nicht nur Kundendaten, sondern auch Daten der Mitarbeitenden und Daten Dritter. Unter Datensicherheit versteht man die Vertraulichkeit (Schutz vor unautorisiertem Zugriff), die Integrität (Schutz vor beabsichtigten oder unbeabsichtigten Veränderungen), die Verfügbarkeit (Gewährleistung des ständigen Zugriffs auf die Daten) und die Kontrollierbarkeit (Protokollierung der Zugriffe und Verarbeitung). Datensicherheit hat also zum Ziel, beliebige Daten vor Schäden wie Manipulation und Nicht-Verfügbarkeit zu schützen. Hierzu zählen unter anderem Aspekte wie die physische Sicherheit, der Schutz vor unbefugten Fremdzugriffen, der Schutz vor unbefugten internen Zugriffen, die Verschlüsselung der Kommunikation, die Datensicherung (Backup), sowie auch das zeitnahe Einspielen von Software- und Sicherheits-Updates.“

Die relevanten Auszüge der Richtlinie befinden sich im Anhang A.7.

# 4 Realisierungsphase

In diesem Teil der Dokumentation wird das Projekt umgesetzt. Die Umsetzung ist in mehrere Teilschritte unterteilt.

## 4.1 Beschaffung von Hard- und Software

Da die Entscheidung getroffen wurde die Umsetzung auf einer virtuellen Maschine durchzuführen ist es nicht nötig neue Hardware zu beschaffen. Die benötigte VM wird auf dem Virtualisierungscluster der Produktions-IT bereitgestellt, auf dem die Open-Source-Virtualisierungsplattform Proxmox VE betrieben wird. Der Erstellungsprozess wird im Anhang A.9 beispielhaft anhand von Screenshots dargestellt.

## 4.2 Erweiterung des lokalen Update-Repositories

Viele Systeme der Produktions-IT am Standort Oldenburg sind aus Sicherheitsgründen nicht direkt mit dem Internet verbunden und können daher keine beliebigen Pakete herunterladen, installieren oder aktualisieren. Dies ist ein notwendiger Schritt, um die Systeme vor Attacken aus dem Internet zu schützen und damit weiter abzusichern. Leider bringt dieser Schritt auch Probleme mit sich, da zum Beispiel Systemupdates mit viel Arbeitsaufwand verbunden sind. Um diesem Problem entgegenzuwirken, wurde in den letzten Jahren ein lokales Update-

Repository[4] eingerichtet. Updates werden vom Administrator zuerst in dieses zentrale Repository geladen und getestet. Nach erfolgreicher manueller Freigabe können sich die mit dem Repository verbundenen Systeme anschließend die getesteten Updates installieren. Da es sich bei diesem Repository um ein Debian-basiertes Repository handelt, werden die Updates auf den Debian Clients mit APT, dem Advanced Packaging Tool[5], aktualisiert. Mit dem Befehl $apt\ update$ werden die Paketlisten aktualisiert und mit $apt\ upgrade$ die Pakete vom Repository heruntergeladen.

Da es auf lange Sicht das Ziel ist alle relevanten Systeme der Produktions-IT in das zentrale Loggingsystem zu integrieren, wird auf diesen Systemen neue Software benötigt, speziell die von Elastic entwickelte Software „Filebeat“. Ich habe eine Anfrage an den Administrator gestellt, der das Update-Repository verwaltet, um das Filebeat Paket in das Update-Repository aufzunehmen. Nach einem kurzen Fachgespräch von 30 Minuten zu meiner Anfrage hat der Kollege das Paket in etwa 30 Minuten bereitgestellt, sodass ich es auf den relevanten Systemen der Produktions-IT installieren konnte. Die aufgewendete Arbeitszeit des Administrators wird in den Projektkosten später berücksichtigt.

## 4.3 Installation des Elastic-Stacks

Der Elastic-Stack wird für dieses Projekt auf einer Linux Maschine mit der Debian Distribution installiert. In den folgenden Abschnitten wird Debian 11 zuerst auf der vorher erstellten virtuellen Maschine installiert. Anschließend werden die drei für das Logging-System benötigten Komponenten Elasticsearch, Kibana und Logstash konfiguriert. Zum Abschluss wird der Log-Shipper Filebeat auf einem System der Produktions-IT installiert, um dort Log-Dateien zu erfassen und an das Logging-System zu schicken.

### 4.3.1 Installation von Debian 11

Da als Basis des Elastic-Stacks eine virtuelle Maschine mit Debian dienen soll, muss zuerst Debian auf der neu erstellten virtuellen Maschine installiert werden. Da Debian 11 in der Produktions-IT fast ausschließlich zum Einsatz kommt, liegt bereits ein passendes Installationsabbild auf dem Virtualisierungscluster bereit und kann als virtuelle Installations-CD zur Maschine hinzugefügt werden. Ich habe zur Installation ein sogenanntes Minimal-Abbild verwendet, bei dem lediglich die Kernkomponenten des Betriebssystems installiert werden.

Da für die Installation und Administration des Elastic-Stacks lediglich die Konsole benötigt wird, installiere ich eine Version ohne GUI. Dies spart sowohl Speicherplatz als auch Rechenkapazitäten ein. Bei der Installation von Debian wird ein Benutzers samt Kennwort werden noch das root-Passwort festgelegt und die Partitionierung der Festplatte, Auswahl der Zeitzone, Sprache und des Tastatur-Layouts ausgewählt. Am Ende der Installation lassen sich noch einige Zusatzkomponenten installieren. Hier wird alles abgewählt außer „SSH server“ und „standard system utilities“. Anschließend wird die Installation fertiggestellt. Der Installer fordert den Nutzer auf, das Installations-Abbild aus dem virtuellen CD-Laufwerk zu entfernen und die Maschine daraufhin neu zu starten. Dies schließt die Installation von Debian 11 ab.

[4] Debian Repository. URL: https://wiki.debian.org/DebianRepository (abgerufen am 06.05.2022)
[5] Advanced Packaging Tool. URL: https://wiki.ubuntuusers.de/APT/ (abgerufen am 10.05.2022)

### 4.3.2 Installation von Elasticsearch

Als Erste der drei Komponenten des Elastic-Stacks muss Elasticsearch installiert werden, da dieses später als Referenz für die Verbindung von Logstash als auch Kibana verwendet wird. Die Installation und Funktion von Elasticsearch benötigen das Tool $wget$. Mit $wget$ lassen sich Dateien über die Kommandozeile direkt von FTP-, HTTP- und HTTPS-Servern herunterladen. Die Installation von $wget$ erfolgt über den Befehl $sudo\ apt\ install\ wget$. $wget$ ist standardmäßig in der Paketverwaltung von Debian enthalten.

Anschließend kann mit der Installation von Elasticsearch fortgefahren werden. Da das Debian 11 Repository bereits in Schritt 4.3 um den Mirror von Elastic erweitert wurde, kann Elasticsearch problemlos mit dem Befehl $sudo\ apt\ install\ elasticsearch$ installiert werden. Sobald die Installation abgeschlossen ist, schreibt der Installer mehrere Zeilen Text mit Informationen zu Elastic in die Konsole. Diese Informationen müssen dringend abgespeichert werden, da sie einen Token enthalten, um später Kibana einfacher mit Elasticsearch zu verbinden. Ebenfalls enthält die Ausgabe Befehle, um die Elastic-internen Nutzer zurückzusetzen oder Passwörter zu generieren. Die Installation von Elastic ist damit abgeschlossen. Die Konfiguration und das Starten des Dienstes erfolgen in einem späteren Schritt.

### 4.3.3 Installation von Kibana

Kibana wird mit dem Befehl $sudo\ apt\ install\ kibana$ installiert. Nach der Installation muss zunächst nichts Weiteres vorgenommen werden. Die Konfiguration und der Start von Kibana erfolgen in Schritt 4.5 und 4.6.

### 4.3.4 Installation von Logstash

Logstash wird mit dem Befehl $sudo\ apt\ install\ logstash$ installiert. Wie bereits bei Kibana muss auch hier vorerst nichts weiter getan werden. Konfiguration und Start folgen in späteren Schritten.

## 4.4 Konfiguration des ELK-Stacks

In den folgenden Schritten werden Elasticsearch, Kibana und Logstash konfiguriert.

### 4.4.1 Konfiguration von Elasticsearch

Die Konfigurationsdatei für Elasticsearch befindet sich im Verzeichnis $/etc/elasticsearch/$. Die Datei wird mit dem Textbearbeitungsprogramm $nano$ über den Befehl $sudo\ nano\ elasticsearch.\ yml$ geöffnet. In dieser Datei wird unter anderem der Name des Elasticsearch-Clusters festgelegt. Dieser wird benötigt, auch wenn Elasticsearch nur als Single-Node betrieben wird. Auf dem Test-Server wurde das Cluster „ol-elastic“ genannt. Der Parameter $network.\ host$: wird auf $0.0.0.0$ gesetzt. $0.0.0.0$ bedeutet in diesem Kontext, dass Elasticsearch auf allen auf dieser Maschine konfigurierten IPv4 Adressen erreichbar ist. Da lediglich eine IP-Adresse konfiguriert ist, wird Elasticsearch später auch nur über diese erreichbar sein. Die Zeile $http:\ port:\ 9200$ wird auskommentiert. 9200 ist der Standard-Port über den Elasticsearch kommuniziert und dieser Port wird nicht verändert. $path.\ data$ und $path.\ logs$ bleiben auf den Standardwerten. $path.data$ zeigt auf den Pfad $/var/lib/elasticsearch$ und

beschreibt den Pfad an dem Elasticsearch Daten abspeichert. $path.logs$ verweist auf $/var/log/elasticsearch$ und beschreibt den Pfad an dem Elasticsearch eigene Log-Dateien erstellt. Weitere mögliche Konfigurationsparameter sind Optionen zur Einrichtung eines Elasticsearch-Clusters, da in diesem Projekt allerdings lediglich eine Single-Node Lösung eingerichtet wird, werden die Cluster Optionen nicht benötigt.

### 4.4.2 Konfiguration von Kibana

Die Konfigurationsdatei für Kibana befindet sich im Ordner $/etc/kibana/$. Die Datei wird mit dem Befehl $nano\ kibana.yml$ geöffnet. Die für dieses Projekt wichtigen Parameter sind $server.host$, $elasticsearch.hosts$ und $server.port$.

$server.\ port$ gibt den Kommunikationsport an, den Kibana verwenden soll. Hier wird der Standard-Port 5601 verwendet. $server.\ host$ ist die IP-Adresse, unter der Kibana erreicht werden soll. Hier wird, wie bei Elasticsearch, die 0.0.0.0 eingetragen. Damit bindet sich auch Kibana an alle verfügbaren Interfaces. Da wir aber lediglich ein Interface konfiguriert haben, bekommt der Kibana Server damit auch nur eine verwendbare IP-Adresse. Als Letztes wird der Parameter $elasticsearch.\ hosts$ konfiguriert. Dieser gibt an von welchem Elasticsearch-Host Kibana seine Daten bezieht. Hier werden die IP-Adresse und Port des vorher konfigurierten Elasticsearch-Servers eingetragen.

### 4.4.3 Konfiguration von Logstash

Als letzte Komponente wird Logstash konfiguriert. Die Konfigurationsdatei von Logstash selbst muss nicht bearbeitet werden. Die Standardeinstellungen genügen für den Projektumfang völlig. Die relevanten Einstellungen für den Datenimport, Datenexport und das Verarbeiten der Daten werden in separaten Konfigurationsdateien vorgenommen. Dafür wird als Erstes ein Ordner mit dem Namen $conf.\ d$ im Logstash Ordner unter $/etc/logstash/$ erstellt. Als nächstes wird der Pfad dieses Ordners in der Datei $pipelines.\ yml$ konfiguriert. Diese befindet sich ebenfalls im Ordner $/etc/logstash$. Hier wird der Ordner $conf.\ d$ als Quellverzeichnis für weitere Pipelines konfiguriert. Pipelines beschreiben für Logstash Vorgänge, wie Daten importiert, verarbeitet und exportiert werden.

Anschließend werden drei Dateien im Ordner $conf.d$ angelegt: $beats-input.conf$, $filter.\ conf$ und $elasticsearch-output.\ conf$. In $beats-input.\ conf$ wird der Port beschreiben, auf dem Logstash Logdateien von Filebeat empfängt. Der Standardport ist hier Port 5044. In $filter.conf$ wird beschrieben, wie die Dateien empfangen, sortiert und verändert werden. $elasticsearch-output.\ conf$ beinhaltet die Adresse des Elasticsearch-Servers, an den die Daten gesendet werden, damit sie dort für Kibana verfügbar sind. Die Konfigurationsdateien können im Anhang A.3 eingesehen werden.

## 4.5 Starten des ELK-Stacks

Nachdem die Konfigurationen vorgenommen und alle Komponenten installiert wurden, kann der ELK-Stack gestartet werden. Zuerst wird Elasticsearch mit dem Befehl $sudo\ systemctl\ start\ elasticsearch.\ service$ gestartet. Um zu gewährleisten, dass Elasticsearch bei einem Neustart des Betriebssystems automatisch gestartet wird, wird noch der Befehl $sudo\ systemctl\ enable\ elasticsearch.\ service$ ausgeführt. Diese Schritte werden nun sowohl für den Service $kibana.service$ und $logstash.service$ wiederholt. Um zu prüfen, ob die Dienste

gestartet wurden, wird der Befehl $sudo\ systemctl\ status\ \%SERVICENAME$ ausgeführt, wobei $\%SERVICENAME$ den Namen des jeweiligen Dienstes repräsentiert. Als nächstes müssen dem Logging-System Daten zugeführt werden. Dies erfolgt im Schritt 4.7.

## 4.6 Installation von Filebeat

Filebeat ist ein sogenannter Log-Shipper. Ziel des Programmes ist es ausgewählte Log-Dateien an ausgewählte Server zu schicken, um die Log-Dateien dort zu lagern oder sie weiterzuverarbeiten. Da ich bereits im restlichen Projektumfeld mit Produkten von Elastic arbeite erschien es mir sinnvoll auch beim Log-Shipper auf ein Produkt von Elastic zu verwenden. Dazu kommt, dass Filebeat sowohl unter Linux als auch Windows funktioniert und damit alle Betriebssysteme, die in der Produktions-IT eingesetzt werden, abdeckt. Nachdem Filebeat in das Update-Repository aufgenommen wurde (wie in 4.1.2 beschrieben) müssen die Paketquellen auf der lokalen Maschine aktualisiert werden. Dies passiert durch die Eingabe des Befehls: $apt\ update$. Anschließend wird Filebeat mit dem Befehl $sudo\ apt\ install\ filebeat$ installiert. Hierbei ist zu beachten, dass root Privilegien benötigt werden. Das Programm wird in seinem unter Linux üblichen Installationspfad installiert. Die Programmdateien werden in: $/usr/share/filebeat/$ abgelegt, während die Konfigurationsdateien unter: $/etc/filebeat/$ gespeichert werden. In der Datei $filebeat.\ yml$ in $/etc/filebeat/$ werden die Konfigurationen für die Exporte der Log-Dateien vorgenommen. Hier wird konfiguriert, welche Log-Dateien von dieser Maschine gesammelt und verschickt werden sollen, welche Zeilen der Log-Dateien relevant sind und an welchen Server die Log-Dateien geschickt werden sollen. Es lassen sich noch viele andere Parameter konfigurieren, diese sind aber für das gewünschte Projektziel nicht nötig. Auf die wichtigen Aspekte der Konfiguration wird in Kapitel 4.6.1 der Projektdokumentation eingegangen.

### 4.6.1 Konfiguration von Filebeat

Nachdem Filebeat auf dem System installiert wurde, muss Filebeat konfiguriert werden. Hier wird die Konfiguration eines Clients mit dem Namen RestartHub vorgenommen. Die Konfigurationsdatei von Filebeat kann aus mehreren Komponenten bestehen, je nach Nutzungsumfang. Im Anhang A.5 ist die Konfigurationsdatei $filebeat.\ yml$ hinterlegt. In dieser Datei werden die Dateipfade, die geloggt werden sollen, eingetragen. – $type$: $log$ gibt die Art von Dateien an, die berücksichtigt werden sollen. $enabled$: $true$ bestätigt, dass dieser Input auch tatsächlich aktiv ist. Unter $paths$ wird der Pfad definiert, in dem Filebeat nach den Dateien sucht. Auf diesem Client wurde eine spezifische Datei definiert, nämlich $RestartHub.\ log$. Es kann hier aber auch mit Wildcards gearbeitet werden, um mehrere Dateien in einem Input zusammenzufassen, zum Beispiel können bei einer Pfadangabe von $/var/log/*.\ log$ alle Dateien im Ordner $/var/log/$ gesammelt werden, die die Dateiendung $.log$ haben.

Mit dem $include\_lines$ Parameter kann bereits vorab bestimmt werden, welche Logzeilen Filebeat überhaupt beachten soll. In diesem Beispiel werden nur Zeilen berücksichtigt, die die Wörter $FATAL$ und $JdfHubController$ enthalten. Grund dafür ist, dass die Log-Datei regelmäßig über 100.000 Zeilen umfasst und ein Großteil dieser Zeilen für die Fehlersuche irrelevant sind. In Zeilen, die denen die beiden oben genannten Schlüsselwörter enthalten sind, finden sich spezifische Informationen, wie zum Beispiel eine Kundenauftragsnummer, die bei der Fehlersuche enorm wichtig sind. Anhand des $fields$ Parameters erhalten Log-Dateien des jeweiligen Filebeat-Inputs ein Tag, anhand sie identifiziert werden können. So lassen sich die

Log-Zeilen, die aus der $RestartHub.\ log$ übertragen wurden, später mit einem einfachen Vergleich auf dem Logstash Server selektieren.

Der Parameter $output.\ logstash$ wird auskommentiert, das heißt die #-Zeichen werden entfernt. Damit wird der Output an den Logstash Server aktiviert. Ebenfalls wird die $hosts$ Zeile in der Logstash Rubrik auskommentiert und [„127.0.0.1“] wird durch [„$Logstash\ -\ IP$“] ersetzt. Damit ist der Filebeat-Log-Shipper auf dem Client konfiguriert.

# 5 Testphase

Im Folgenden werden die installierten Komponenten getestet und die Konfigurationen auf Fehler überprüft.

## 5.1 Status der Dienste überprüfen

Bevor die Datenimports geprüft werden, wird der Status aller relevanten Komponenten getestet. Auf dem Server werden die Befehle $sudo\ systemctl\ status\ elasticsearch.service$, $sudo\ systemctl\ status\ kibana.\ service$ und $sudo\ systemctl\ status\ logstash.\ service$ ausgeführt, um den entsprechenden Status zu prüfen. Auf der Maschine, von der Log-Dateien geloggt werden sollen, wird der Befehl $sudo\ systemctl\ status\ filebeat.\ service$ ausgeführt. Da alle Komponenten beim Start ihre jeweilige Konfigurationsdatei einlesen und überprüfen, erspart dies die Notwendigkeit jede einzelne Konfigurationsdatei erneut händisch zu prüfen: Gäbe es Syntax-Fehler in den Dateien würde dies in den Statusmeldungen des entsprechenden Dienstes ausgegeben werden. Da alle Komponenten erfolgreich gestartet sind und keine Fehlermeldungen ausgeben kann als nächstes der Datenimport in Kibana überprüft werden.

## 5.2 Datenimports überprüfen

Nachdem alle Komponenten des ELK-Stacks gestartet und konfiguriert und auch Filebeat erfolgreich eingerichtet wurde, kann Kibana auf Daten überprüft werden. Kibana wird über einen Webbrowser aufgerufen. Die Adresse setzt sich aus der in der Konfiguration festgelegten IP-Adresse und Port zusammen, z.B. 192.168.10.200:5601. Nach der Eingabe der IP-Adresse und Port im Browser folgt die Anmeldemaske. Für die Anmeldung in dieser wird der Standardnutzer $elastic$ verwendet. Das Passwort für diesen Benutzer wurde nach der Installation von Elasticsearch auf dem Bildschirm ausgegeben. Sollte das Passwort nicht mehr aufzufinden sein, kann ein neues, zufälliges Passwort generiert werden. Dafür wird auf dem Server in den Pfad $/usr/share/elasticsearch/bin/$ navigiert und führt die Datei $elasticsearch-reset-password$ mit dem Parameter $-u\ elastic$ aus. Daraufhin wird das neue Passwort in der Konsole ausgegeben.

War die Anmeldung erfolgreich wird der User von der Elastic-Oberfläche begrüßt.[6] Um zu prüfen, ob Log-Dateien erfolgreich ans System ermittelt werden, wird oben links auf den Menü-Button geklickt und im Bereich „Analytics“ der Menüpunkt „Discover“ gewählt. Dies führt den Nutzer in die Log-Übersicht.[7] Hier findet der User in der oberen Hälfte einen Zeitstrahl, in dem grüne Balken visuell darstellen, ob und wie viele Einträge geloggt wurden. Darunter findet sich eine zeitlich sortierte Liste der einzelnen Log-Ereignisse (von Neustem/Ältestem). Ein Klick auf

[6] Siehe Anhang A.10: Oberfläche Elastic
[7] Siehe Anhang A.10.1: Übersicht Log-Dateien

diese Log-Ereignisse klappt diese aus und zeigt dem Nutzer alle Details zu diesem Ereignis.[8] Hier finden sich Informationen wie Hostname, Host-IP-Adresse, Betriebssystem sowie Uhrzeit und Quell-Logdatei. Die eigentliche Nachricht, die aus der Log-Datei gelesen wurde, befindet sich im Feld $message$. Über die Schaltfläche „Add filter“ lassen sich die Suchergebnisse vorab filtern, zum Beispiel nach Hostnamen. So lassen sich später Systeme vom selben Typ oder Standort einfach Gruppieren, um die Suche einzuschränken.[9] Im Beispiel im Anhang 10.2 werden z.B. nur Logs von Systemen berücksichtigt, deren Hostname mit $ol - vm$ beginnt. Die Filter bleiben aktiviert, bis sie manuell wieder entfernt werden. Da Log-Dateien im System eingegangen sind und auch die Filterfunktion Suchergebnisse liefert ist der Funktionstest damit bestanden.

### 5.3 Fehlerkorrektur

Ein Tippfehler in der Konfigurationsdatei von Filebeat auf dem Client RestartHub hat dazu geführt, dass eine notwendige Art von Ereignis nicht geloggt wurde. Zeilen, die das Wort „FATAL“ enthalten, sollten dringend geloggt werden. In der Konfigurationsdatei wurde aber versehentlich „FATL“ geschrieben.

## 6 Abschlussphase

### 6.1 Berechnung der Kosten für den physischen Server

Im Folgenden finden sich die in Tabelle 5 kalkulierten Kosten für den physischen Server für das erste Jahr. Es wurde mit einem durchschnittlichen Stromverbrauch von 60W gerechnet. Bei den Kosten handelt es sich, ausgenommen von den Serverkosten und Personalkosten, um fiktive Werte. Die Kosten für den Einbau sowie für den Server selbst wurden in den laufenden Kosten in Tabelle 3 nicht berücksichtigt.

| **Posten** | **Anzahl** | **Einzelpreis** | **Gesamtpreis** |
|---|---|---|---|
| 1HE Intel Single-CPU RI1101H-XE Server | 1 Stück | 2.842,17€ | 2.842,17€ |
| Einbau durch Administrator | 1 Stunde | 85,00€ | 85,00€ |
| Stromkosten | 525,6 kWh | 0,2139€ | 112,42€ |
| Wartung | 9 Stunden | 85,00€ | 765,00€ |
| Sonstige Kosten | - | 511,15€ | 511,15€ |
| **Summe** | | | **4.315,74€** |

*Tabelle 5 Kostenkalkulation physischer Server*

### 6.2 Berechnung der Kosten für die virtuelle Maschine

Für die Berechnung der Kosten der virtuellen Maschine wurden verschieden Faktoren berücksichtigt, wie zum Beispiel die Anschaffungskosten der verschiedenen Virtualisierungshosts des Clusters, der Storage- und Backupsysteme sowie die Arbeitskosten für die Administration, Stromkosten und Andere. Bei den Kosten in Tabelle 6 handelt es sich um Schätzkosten, die von der Fachabteilung Produktions-IT für das Virtualisierungscluster kalkuliert wurden.

[8] Siehe Anhang A.8: Log-Query
[9] Siehe Anhang A.10.2: Filteroptionen

| Anschaffungskosten | Anzahl | Einzelpreis | Gesamtpreis |
|---|---|---|---|
| Virtualisierungshost | 5 | 17.000€ | 85.000,00€ |
| Blockstoragesystem | 1 | | 70.000,00€ |
| Backupstoragesystem | 1 | | 10.000,00€ |
| | | Anschaffungskosten gesamt | **165.000,00€** |

*Tabelle 6 Beschaffungskosten Virtualisierungscluster*

| Nebenkosten | Anzahl | Einzelpreis | Gesamtpreis |
|---|---|---|---|
| Proxmox Support | 6 | 1.062,00€ | 6.372,00€ |
| Rechenzentrum | 16 HE | 120€ | 1.920,00€ |
| Stromkosten | | | 2.629,00€ |
| Administration | | | 3.825,00€ |
| | | Nebenkosten pro Jahr | **14.746,00€** |

*Tabelle 7 Nebenkosten Virtualisierungscluster*

| | | |
|---|---|---|
| Anschaffungskosten | 135.000€ ÷ 2 Jahre | 67.500,00€ |
| Nebenkosten | | 14.746,00€ |
| | Kosten des Clusters pro Jahr | **82.246,00€** |

*Tabelle 8 Gesamtkosten Virtualisierungscluster*

$$82.246€/Jahr \div 240\,VM's\ auf\ dem\ Cluster = 343€\ pro\ VM\ pro\ Jahr$$

$$343€ \div 12\ Monate = 29€\ pro\ Monat\ pro\ VM$$

$$343€ \div 365{,}25\ Tage = 0{,}93€\ pro\ Tag\ pro\ VM$$

Die Pauschalkosten für den Betrieb einer VM für einen Tag betragen etwa 0,93€. Dies beinhaltet Beschaffungskosten, Stromkosten, Wartung und Support. Aktuell werden auf dem Cluster etwa 240 virtuelle Maschinen betrieben. Die Kosten der VM pro Jahr, Monat und Tag wurden auf volle Eurowerte aufgerundet.

## 6.3 Personalkosten

Während der Umsetzung sind Personalkosten in zwei weiteren Abteilungen angefallen. Die genaue Stundenanzahl sowie Stundensatz sind in der Tabelle 9 zu finden. Bei den Stundensätzen wird von einem Mitarbeiter der Tarifgruppe K4 mit weniger als 4 Jahren Betriebszugehörigkeit ausgegangen, was bei CEWE etwa den Durchschnitt im Bereich der Produktions-IT abbildet.

| Aufgaben | Abteilung | Stunden | Stundensatz | Gesamtpreis |
|---|---|---|---|---|
| Repository-Erweiterung | Produktions-IT | 1 | 85€ | 85,00€ |
| DNS-Eintrag | Netzwerk | 0,5 | 85€ | 42,50€ |
| | | | Einmalige Personalkosten | **127,50€** |

*Tabelle 9 Personalkosten*

## 6.4 Gesamtprojektkosten

Die Gesamtprojektkosten setzen sich zusammen aus 35 Arbeitsstunden für die Projektumsetzung, voraussichtliche Kosten für die virtuelle Maschine über zwei Jahre und die Personalkosten der verschiedenen Abteilungen. Dazu kommen etwa zwei Stunden monatlich für die Erfassung neuer Systeme, die geloggt werden sollen, und die dazugehörige Installation- und Konfiguration von Filebeat. Zusätzlich muss mit etwa einer Stunde monatlich für die Administration und Pflege des Systems gerechnet werden.

| Gesamtkosten Umsetzung | Gesamtpreis |
|---|---|
| Arbeitszeit Kai Müller 35 Stunden á 45,42€ | 1.590,00€ |
| Personalkosten | 127,50€ |
| Kosten der VM über 2 Jahre | 678,90€ |
| Erfassung neuer Systeme für das Logging 36 Stunden á 85€ | 3.060,00€ |
| Administration des Logging-Systems 24 Stunden á 85€ | 2.040,00€ |
| Erwartete Gesamtkosten über Projektlaufzeit | **7.496,40€** |

*Tabelle 10 Erwartete Gesamtkosten über die Projektlaufzeit*

## 6.5 Amortisierung

Der aktuelle Arbeitsablauf für die Fehlersuche anhand von Log-Dateien dauert im Schnitt etwa 15 Minuten. Fehler im Monitoring wahrnehmen, entsprechendes Host-System anhand des Namens erkennen und im Anschluss die IP-Adresse oder den Hostnamen im Virtualisierungscluster suchen. Daraufhin remote auf das System zugreifen und dann die Log-Datei im entsprechenden Ordner finden und öffnen und die Fehlersuche beginnen. Der neue Workflow geht nach der Wahrnehmung des Fehlers im Monitoring direkt zu Elastic über. Dabei wird die Logging-Übersicht aufgerufen und im Suchfeld nach $host.\,hostname{:}\ ol - vm - \%KÜRZEL$ gesucht, wobei $\%KÜRZEL$ für eine Kennung steht, die alle Maschinentypen haben. Es wird mit einer erwarteten Reduzierung des Arbeitsaufwandes von 50% bis 60% gerechnet. Wird von 15 Minuten und zwei Fehlersuchen über Log-Dateien pro Tag und von etwa 21 Arbeitstagen pro Monat ausgegangen, ergibt das 630 Minuten oder 10,5 Stunden im Monat. Dies ergibt bei einem Stundensatz von 85€ eine Summe von 892,50€ im Monat. Eine Arbeitsaufwandreduzierung von 60% bedeutet einen monatlichen Arbeitsaufwand von 252 Minuten oder 4,2 Stunden. Dies ergibt eine Summe von 357€. Ausgehend von diesen Werten amortisiert sich das Projekt nach dem 16. Monat. (vgl. Anhang A.12)

## 6.6 Fazit

Innerhalb der 35 Projektstunden konnte ich viele verschiedene Aufgaben im Projektumfeld bearbeiten. Die Erweiterung des Update-Repositories und das Anlegen der DNS-Einträge hätte ich mir gerne im Detail angeschaut und gegebenenfalls selbst umgesetzt. Aufgrund der anhaltenden Covid-19 Pandemie und den firmeninternen Abstandsregeln war es mir jedoch nicht möglich mich über einen längeren Zeitraum mit den entsprechenden Kollegen zusammenzusetzen. Daher habe ich die Entscheidung getroffen, die Arbeitszeit der entsprechenden Kollegen wie die von einem Dienstleister zu kalkulieren. Dies sind die Personalkosten, die in Kapitel 6.3 errechnet wurden.

Bei der Zeitplanung gab es einige, kleinere Abweichungen. So habe ich über den Datenschutzaspekt des Projekts erst nachgedacht, als ich bereits Software verglichen habe. Die Zeit dafür musste ich daher an anderen Teilen einsparen. Dies war aber kein Problem, da anderen Bereiche weniger Zeit benötigten als geplant, wie zum Beispiel die Aufnahme des Ist-Zustandes. Ebenso waren die Installation und Konfiguration schneller erledigt, als erwartet. Da für den Einsatz eine VM gewählt wurde, fiel die Beschaffung eines physischen Servers weg. Ebenso musste keine Software gekauft werden, da komplett auf Open-Source Software gesetzt wurde. Würde ich das Projekt noch einmal durchführen, würde ich beim nächsten Mal genauer festhalten, welche Erwartungen der Projektverantwortliche an das System hat, da dies im Projektverlauf öfter unklar war. Die Projektdauer von 35 Stunden konnte aber schlussendlich eingehalten werden.

# A Anhang

## A.1 Glossar und Abkürzungsverzeichnis

| Nummer | Begriff | Erklärung |
|---|---|---|
| 1 | VM | Virtuelle Maschine |
| 2 | On-Premise | Betrieb eines Gerätes auf dem Firmengelände |
| 3 | Logs/Log-Dateien | Automatisch erstellte Dateien, die Informationen zum Betrieb einer Soft- oder Hardware beinhalten |
| 4 | SSH | Secure Shell – kryptografisches Netzwerkprotokoll, um in einem unsicheren Netzwerk sicher auf Computer zu-zugreifen |
| 5 | VNC | Virtual Network Computing – Verbindung auf die grafische Oberfläche eines entfernten Computers |
| 6 | Log-Shipper | Programm, dass Log-Dateien automatisiert von einem Computer zu einem anderen transportiert |
| 7 | Cluster | Verbund von vernetzten Computern, meist Server |
| 8 | DNS | Domain Name System |
| 9 | Parsen | Maschinelle Verarbeitung und Analyse von Daten |
| 10 | YAML | YAML ain't Markup Language, an XML angelehnte Auszeichnungssprache die oft für Konfigurationsdateien verwendet wird |
| 11 | VPN | Virtual Private Network. Dienst, der aus dem WAN einen Tunnel in ein privates Netzwerk aufbaut |
| 12 | WAN | Wide Area Network, das Internet |
| 13 | LAN | Local Area Network. Ein lokales und privates Netzwerk, z.B. internes Firmennetzwerk |

## A.2 Angebot des Servers von Thomas Krenn



THOMAS KRENN®

Tel. +49 8551 9150 0
Fax +49 8551 9150 55
e-mail: info@thomas-krenn.com

Thomas-Krenn.AG, Speltenbach-Steinäcker 1, D-94078 Freyung

| | |
|---|---|
| Warenkorb Nr. | 217852 |
| Datum | 14.04.2022 |

**IHK Abschlussprojekt**

| POS | ARTIKEL | MENGE | EINZELPREIS | GESAMTPREIS |
|---|---|---|---|---|
| 01 | **1HE Intel Single-CPU RI1101H-XE Server *** | 1 | 590,00 €<br>~~869,00 €~~ | 590,00 € |
| | Essential - Paket:<br>- Hotline MO-FR (07.00 - 22.30 Uhr)<br>- Hardware-Austausch NBD | | | |
| | Supermicro Mainboard X11SCL-IF | 1 | 25,00 | 25,00 |
| | 4x SATA-3 + 1x M.2 | 1 | 0,00 | 0,00 |
| | 2x 1Gbit/s LAN onBoard (Intel I210AT) | 1 | 0,00 | 0,00 |
| | Integriertes IPMI onboard (dedizierte Netzwerkkarte) | 1 | 0,00 | 0,00 |
| | Full Remote Management (KVM over LAN, IPMI 2.0) inkl. Managementsoftware, DHCP Konfiguration | 1 | 0,00 | 0,00 |
| | Intel Xeon E-2236 6-Core 3,4GHz 12MB 8GT/s [**Hinweis**: Nur noch begrenzte Stückzahlen verfügbar!] | 1 | 319,00 | 319,00 |
| | 64 GB (2x 32GB) ECC ATP DDR4 3200 RAM (Premium) | 1 | 515,00 | 515,00 |
| | 1,92 TB SATA III Samsung SSD 3D-NAND TLC 2,5" (PM893) | 2 | 451,00 | 902,00 |
| | SATA Kabel 29cm (gerade Stecker) | 2 | 10,00 | 20,00 |
| | 200 Watt energiesparendes Netzteil (80plus Gold) | 1 | 0,00 | 0,00 |
| | Keine Kühlung für Zusatzkarten | 1 | 0,00 | 0,00 |
| | Keine Linux Vorinstallation | 1 | 0,00 | 0,00 |
| | Keine Windows Vorinstallation | 1 | 0,00 | 0,00 |
| | Essential-Paket (D): 24 Monate | 1 | 0,00 | 0,00 |

THOMAS KRENN®

Tel. +49 8551 9150 0
Fax +49 8551 9150 55
e-mail: info@thomas-krenn.com

| | | | | |
|---|---|---|---|---|
| Energiewerte für diese Serverkonfiguration pro Gerät | | | | |
| Maximale Leistungsaufnahme | 121 W | | | |
| British Thermal Unit | 415 BTU/h | | | |
| | Pos. 01 | 1 | 2.371,00 € | 2.371,00 € |

| | |
|---|---|
| Zwischensumme | 2.371,00 € |
| **Optional** zu einer monatlichen Leasingrate » 60 Monate | 46,81 € |
| 0,4% Transportversicherung (von 2.371,00 €) | 9,48 € |
| Verpackungskosten | 7,90 € |
| Nettosumme | 2.388,38 € |
| + 19% MwSt. | 453,79 € |
| **Gesamtsumme (zzgl. Versandkosten)** | **2.842,17 €** |

## A.3 Konfigurationsdateien Logstash

*beats – input.conf*

```
input {
 beats {
    port => [redacted]
 }
}
```

*filter.conf*

```
filter {
      if [fields][sourcefiletype] == "restarterhub" {
        grok {
          match => { "message" => "%{TIMESTAMP: timestamp} %{HOST:hostname} %{GREEDYDATA:restarterhub_message}" }
          add_field => [ "received_at", "%{@timestamp}" ]
          add_field => [ "received_from", "%{host}" ]
        }
        restarterhub_pri { }
        date {
          match => [ "restarterhub_timestamp", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
        }
      }
      if [fields][sourcefiletype] == "jabien" {

      }
    }
```

*elasticsearch – output.conf*

```
output {
 elasticsearch {
   hosts => ["[redacted]"]
   manage_template => false
   index => "%{[@metadata][beat]}-%{[@metadata][version]}-%{+YYYY.MM.dd}"
 }
}
```

## A.4 Konfigurationsdateien Elasticsearch und Kibana

*elasticsearch. yml*

```
#
# ---------------------------------- Cluster -----------------------------------
#
# Use a descriptive name for your cluster:
#
cluster.name: [redacted]

# ------------------------------------ Paths ------------------------------------
# Path to directory where to store the data (separate multiple locations by comma):
path.data: /var/lib/elasticsearch
# Path to log files:
path.logs: /var/log/elasticsearch
# ---------------------------------- Network -----------------------------------
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
network.host: 0.0.0.0
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
http.port: [redacted]
```

*kibana. yml*

```
# Kibana is served by a back end server. This setting specifies the port to use.
server.port: [redacted]

# Specifies the address to which the Kibana server will bind. IP addresses and host names are both valid values.
# The default is 'localhost', which usually means remote machines will not be able to connect.
# To allow connections from remote users, set this parameter to a non-loopback address.
server.host: "0.0.0.0"
# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["http://XXX.XXX.XXX.XXX:[redacted]"]

# Kibana uses an index in Elasticsearch to store saved searches, visualizations and
# dashboards. Kibana creates a new index if the index doesn't already exist.
#kibana.index: ".kibana"

# The default application to load.
#kibana.defaultAppId: "home"
```

## A.6 Konfigurationsdatei Filebeat

*filebeat. yml*

```
# ============================== Filebeat inputs ===============================
filebeat.inputs:
- type: log
  # Change to true to enable this input configuration.
  enabled: true
  # Paths that should be crawled and fetched. Glob based paths.
  paths:
    - /home/[redacted]
  include_lines:
    - 'FATAL'
    - 'JdfHubController'
  fields:
    sourcefiletype: [redacted]
  fields_under_root: false

# ------------------------------ Logstash Output -------------------------------
output.logstash:
  # The Logstash hosts
  hosts: ["XXX.XXX.XXX.XXX:5044"]
```

## A.7.1 IT-Sicherheitsrichtlinie der CEWE Stiftung & Co. KGaA

### 2.2 Grundlegende Verhaltensregeln

a. **Bei der Benutzung der IT-Systeme und der IT-Applikationen in unserem Unternehmen sind Sie an die geltenden Rechtsvorschriften zum Datenschutz und zur Datensicherheit sowie an alle anderen Unternehmensregelungen und Arbeits- / Handlungsanweisungen gebunden.** Sollten Sie unsicher sein, ob und inwieweit Rechtsvorschriften oder Unternehmensregelungen einzuhalten sind, wenden Sie sich an ihren Vorgesetzten.

b. **CEWE trägt Sorge dafür, dass alle betroffenen Mitarbeitenden, die erforderlichen Schulungen und Instruktionen/Anweisungen (Arbeitsanweisungen) erhalten**, welche für den jeweiligen Umgang mit den IT-Systemen und/oder Applikationen erforderlich sind.

c. **Stellen Sie einen IT-Sicherheitsvorfall fest, müssen Sie dieses unverzüglich dem zuständigen (lokalen) IT-Sicherheitsteam melden.**
Hierzu gehört auch der Verlust von Daten (z.B. verlorenes Notebook, gestohlenes Handy, fehlender USB-Stick), bemerkter Datenabzug oder auch nur Verdachtsfälle, die auf einen Sicherheitsvorfall hindeuten. Diese Regelung gilt auch bei Bekanntwerden eines Sicherheitsvorfalls bei einem externen Dienstleister, Handelspartner oder Lieferanten, sofern unternehmensrelevante Daten oder personenbezogene Daten betroffen sein könnten. Das zentrale IT-Sicherheitsteam ist im Zweifel ebenfalls hinzuzuziehen, insbesondere bei drohenden schweren Sicherheitsvorfällen.

d. **CEWE stellt Ihnen ein installiertes und betriebsbereites Arbeitsplatzsystem zur Verfügung, welches den aktuellen Sicherheitsanforderungen entspricht.**
Dazu gehört z.B. eine aktuelle Sicherheitssoftware in Form eines Virenscanners oder besser einer Endpoint Security Software. Diese darf nicht eigenmächtig verändert oder deaktiviert werden. Wenn Sie keine Sicherheitssoftware auf Ihrem IT-Arbeitsplatz vorfinden, wie z.B. Sophos Endpoint Security oder einen aktuellen Virenscanner, ist der lokale IT-Support zu informieren, der Ihnen eine geeignete Sicherheitssoftware installiert.

## 2.3 Umgang mit den Daten des Unternehmens

### 2.3.1 Grundsätzlicher Umgang

a. **Grundsätzlich müssen die Daten des Unternehmens in seiner IT-Infrastruktur verbleiben und dürfen nur auf eigene, bereits freigegebene IT-Systeme oder Datenträger übertragen werden.**
Wenn neue externe IT-Systeme oder Datenträger genutzt werden sollen, müssen Sie diese vorher freigeben lassen. Wenden Sie sich dazu an Ihren Vorgesetzten oder/und an Ihr lokales IT-Sicherheitsteam.

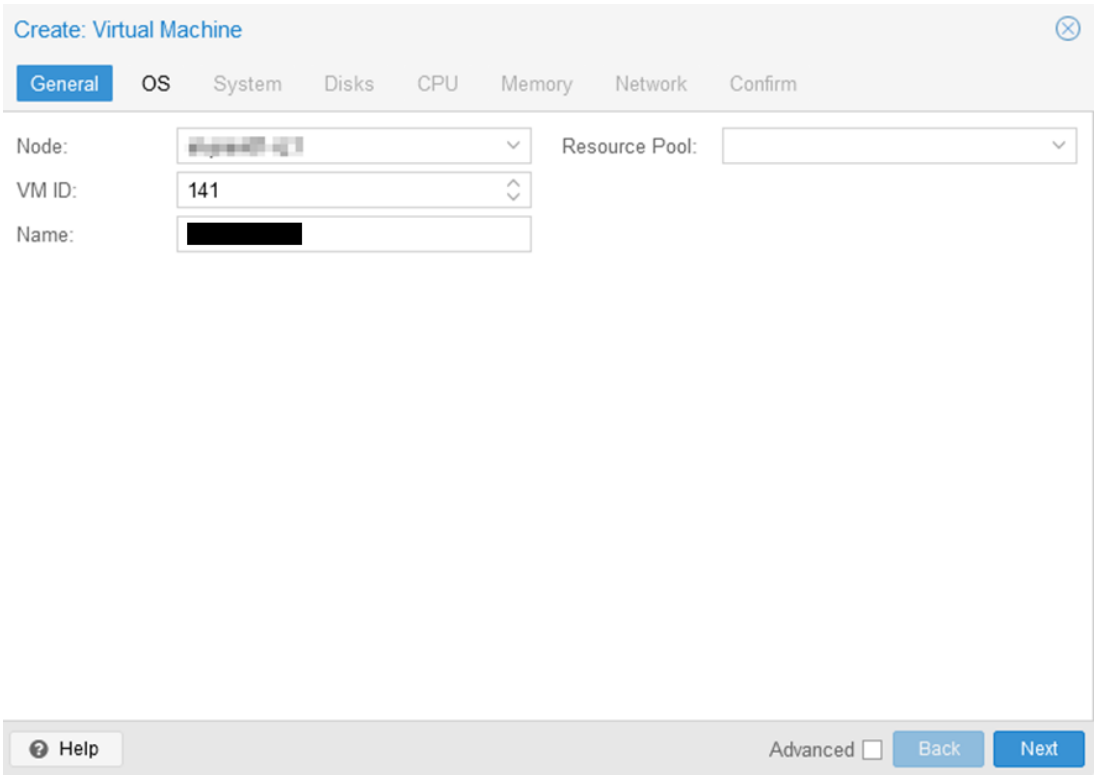
b. **Als Nutzer von IT-Systemen müssen Sie dafür sorgen, dass unbefugte Dritte keinen Zugang zu Daten des Unternehmens erhalten.**
Beispiele hierfür sind:

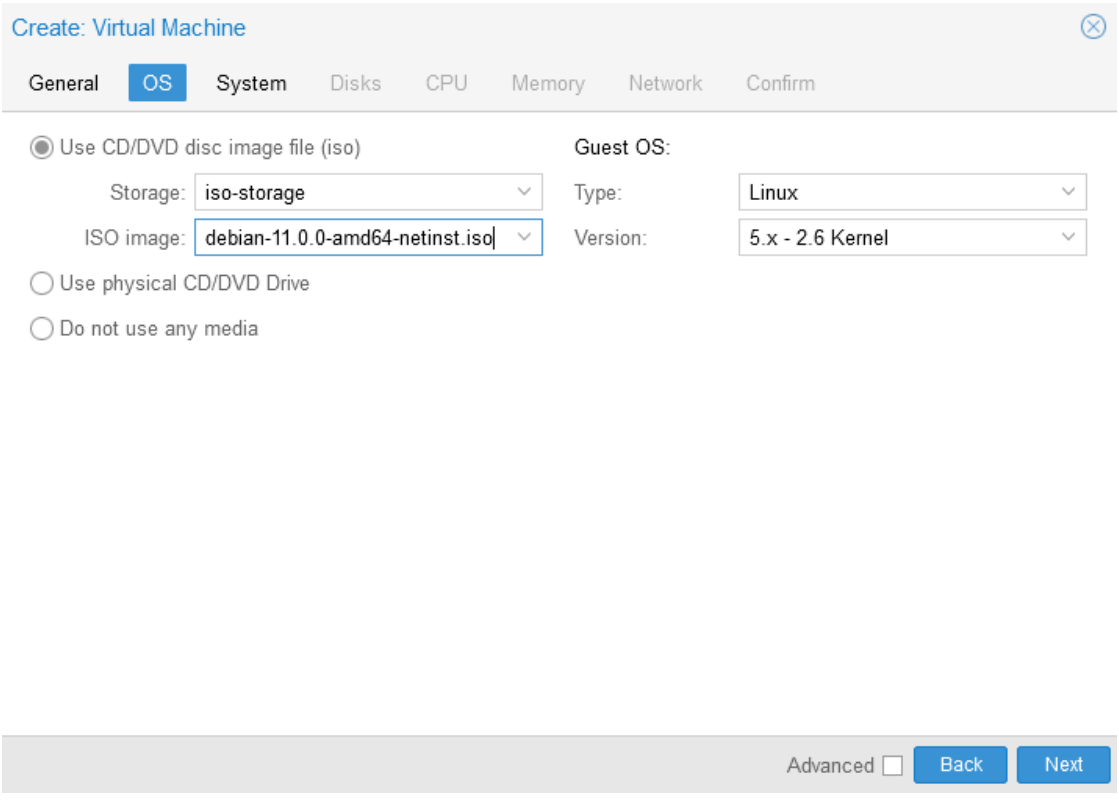
- Nach dem Verlassen des Arbeitsplatzes müssen Sie transportable Medien einschließen, so dass sie nicht von Dritten benutzt werden können.
- Beim Verlassen Ihres Arbeitsplatz-PCs muss das betreffende System gegen Zugriff gesperrt werden, so dass vor der erneuten Nutzung des betroffenen IT-Systems und/oder der Applikation(en) eine Authentifizierung (Benutzername/Passwort) erforderlich wird.
- Informationen in Papierform, wie z.B. Ausdrucke, sind so abzulegen, dass Besucher oder sonstige Dritte keine Kenntnisnahme von den Daten erhalten können.
- Die Benutzung von Mobile Devices, wie z.B. Notebooks und Tablets, im öffentlichem Raum ist so zu gestalten, das Dritte keinen Einblick erhalten (Sichtschutz).
- Dienstliche Telefongespräche mit geschäftskritischem Inhalt sind im öffentlichen Raum so zu gestalten, das Dritte keine Informationen mitbekommen können.
- Vertrauliche Informationen sind stets unter Verschluss zu halten.

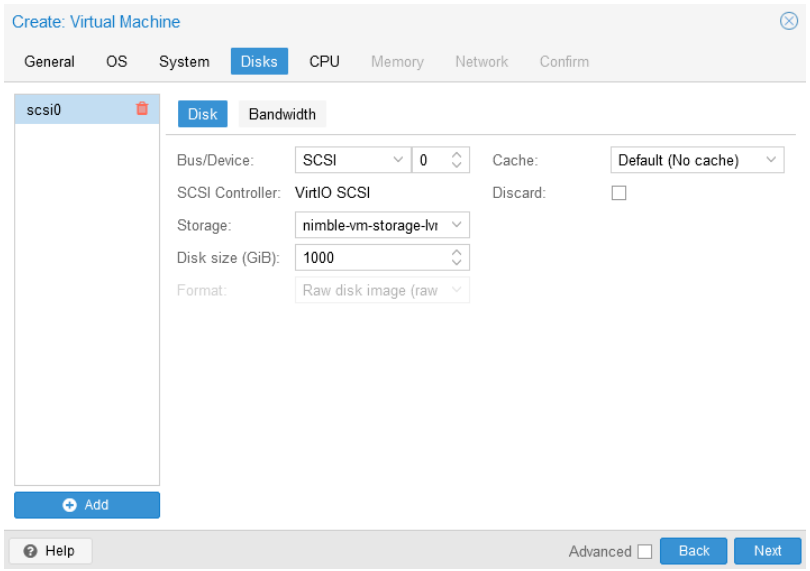
## A.8 Beispiel Elasticsearch-Query

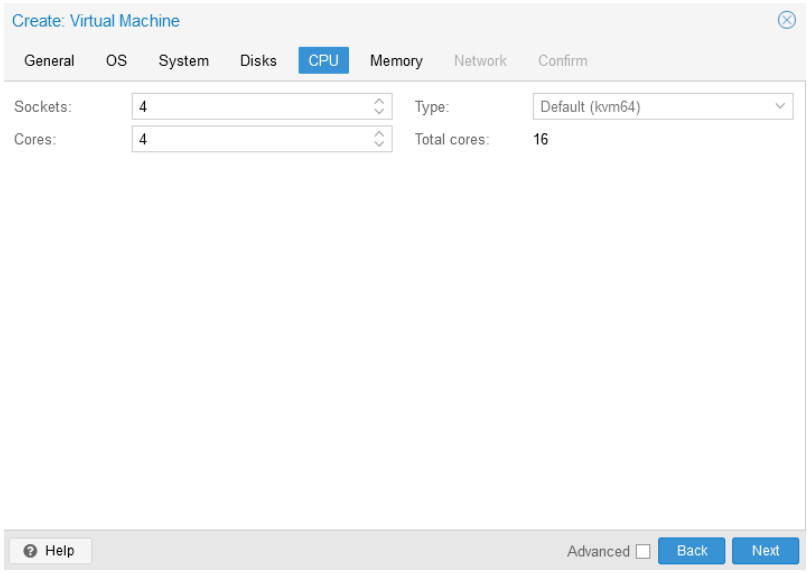
| | |
|---|---|
| host.architecture | x86_64 |
| host.containerized | false |
| host.hostname | [redacted] |
| host.id | [redacted] |
| host.ip | [redacted] |
| host.mac | [redacted] |
| host.name | ol[redacted] |
| host.os.codename | bullseye |
| host.os.family | debian |
| host.os.kernel | 5.10.0-9-amd64 |
| host.os.name | Debian GNU/Linux |
| host.os.platform | debian |
| host.os.type | linux |
| host.os.version | 11 (bullseye) |
| input.type | log |
| log.file.path | /home[redacted] |
| log.offset | 9,567,563 |
| message | 2022-05-05 13:04:51,118 DEBUG JdfHubController |
| tags | beats_input_codec_plain_applied |

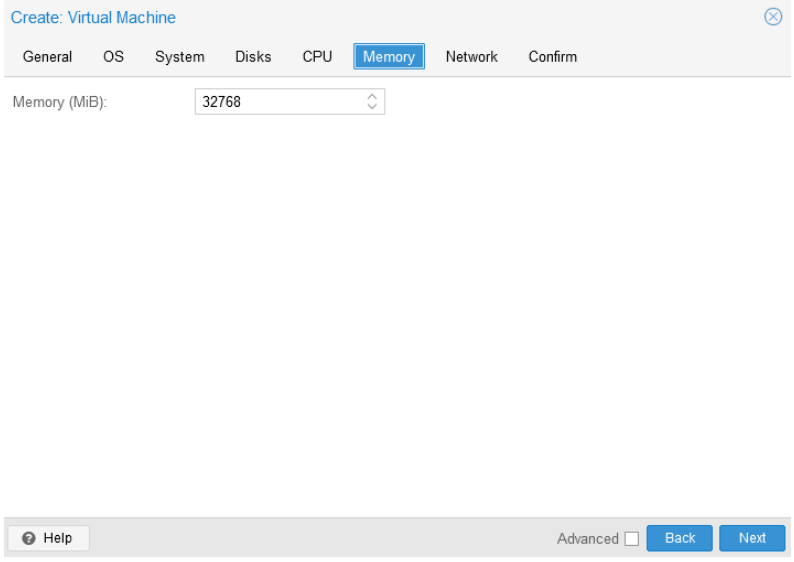
## A.9 Erstellung der VM



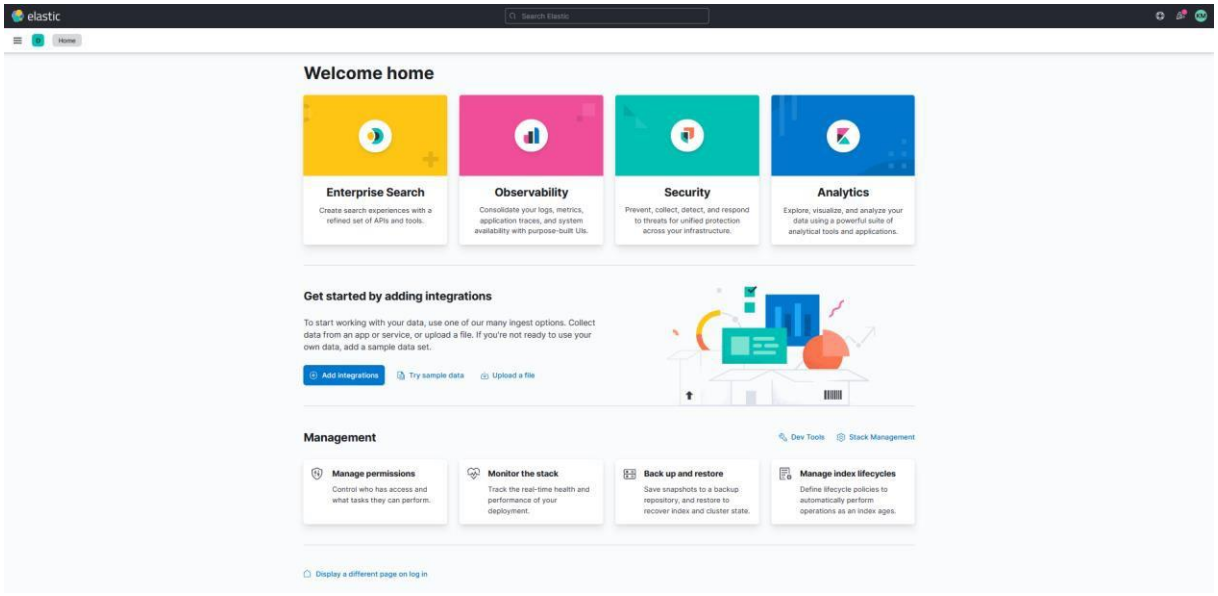




Create: Virtual Machine
General
OS
System
Disks
CPU
Memory
Network
Confirm
scsi0
Disk
Bandwidth
Bus/Device:
SCSI
0
Cache:
Default (No cache)
SCSI Controller:
VirtIO SCSI
Discard:
Storage:
nimble-vm-storage-lv
Disk size (GiB):
1000
Format:
Raw disk image (raw
Add
Help
Advanced
Back
Next

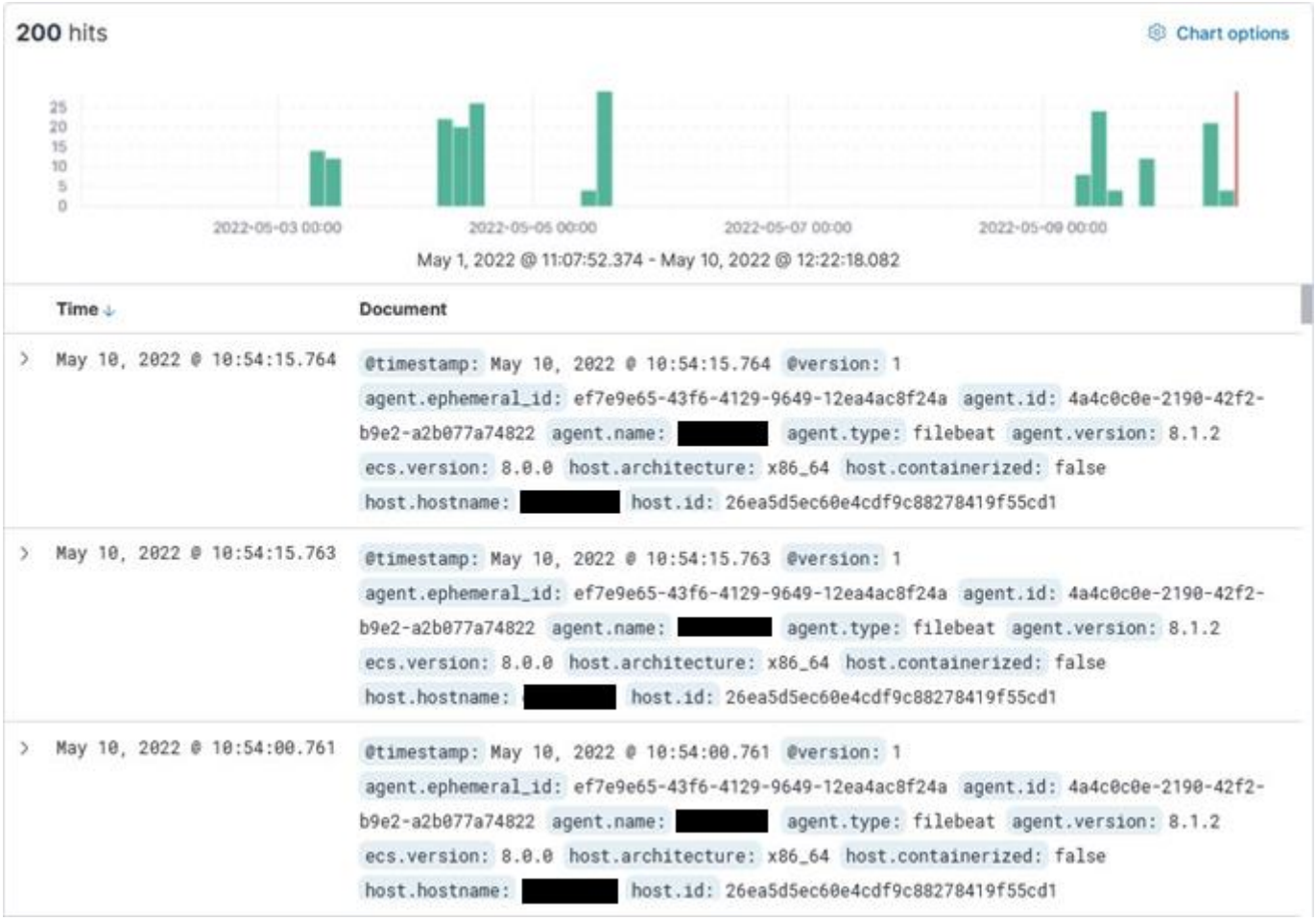

Create: Virtual Machine
General
OS
System
Disks
CPU
Memory
Network
Confirm
Sockets:
4
Type:
Default (kvm64)
Cores:
4
Total cores:
16
Help
Advanced
Back
Next


Create: Virtual Machine
General
OS
System
Disks
CPU
Memory
Network
Confirm
Memory (MiB):
32768
Help
Advanced
Back
Next

## A.10 Elastic Oberfläche




## A.10.1 Übersicht Log-Dateien



## A.11 Dokumentation

### A.11.1 Administratordokumentation

Stand 10.05.2022 Administratordokumentation Autor: Kai Müller

# Administratordokumentation des ELK-Stacks

Zugangsdaten:

Es handelt sich bei der Projektmaschine um ein Linux-System mit Debian 11 ohne grafische Oberfläche. Das System ist erreichbar über SSH auf Port 22 oder über den Proxmox Virtualisierungshost (VM Nummer: 714).

Hostname der VM: [redacted]

IP-Adresse: 192.168.10.100/24

Adresse der Weboberfläche: 192.168.10.100:5601 oder **********

Zugangsdaten für die Weboberfläche:

Benutzername: e******
Passwort: **********

Benutzername: o*********
Passwort: **********

Der User „elastic“ ist der Root-User des Systems. Er verfügt über alle Zugriffsrechte, Rechte neue Nutzer zu erstellen und deren Rechte zu verwalten. Benutzen Sie diesen User, wenn Sie neue Nutzer erstellen möchten.

Der User „ol-logging“ wurde als Anwender-Account angelegt. Dieser User erhält lediglich Zugriff auf die Logging-Übersicht und die Rechte, um einzelne Log-Daten aufzurufen. Der User darf außerdem eigene Suchfilter erstellen.

Zugangsdaten für die VM:

Benutzername: o********
Passwort: **********
Besonderes: User darf Befehle mit sudo ausführen

Benutzername: root
Passwort: **********

Adresse der Elasticsearch API: 192.168.10.100:5600

# Administration des ELK-Stacks

## Anlegen von Filtern für das Log-Parsing

Es besteht, wenn gewünscht, die Möglichkeit, die Log-Dateien noch weiter zu filtern. Dies funktioniert mithilfe eines in Logstash integrierten Plugins namens grok.[10] Grok kann dabei die ins Logstash überführten Log-Dateien parsen und anhand von selbst erstellten Filtern strukturieren. Dies bietet dem Administrator die Möglichkeit, die Log-Dateien auf User-Wünsche zuzuschneiden. Um Filter zu erstellen, sollte die offizielle grok-Dokumentation von Elastic konsultiert werden (siehe Fußnote 7).

## Ändern der Konfiguration der Komponenten des ELK-Stacks:

### Elasticsearch

Die Konfigurationsdatei für Elasticsearch befindet sich im Pfad $/etc/elasticsearch/$ unter dem Namen $elasticsearch.\ yml$ und kann mit einem beliebigen Text-Editor geöffnet werden. Nano ist unter Debian bereits vorinstalliert. Die Struktur der Datei kann im Anhang A.4 eingesehen werden. Die IP-Konfiguration sollte, wenn nicht absolut notwendig, nicht geändert werden, da dies den Betrieb des gesamten Stacks und Filebeat beeinflusst. Cluster-Name und Pfade zur Speicherung der Log-Dateien können geändert werden, ohne die Funktion zu beeinträchtigen. Änderungen an der Konfigurationsdatei sollten allerdings mit Datum und Grund der Änderung dokumentiert werden.

### Kibana

Die Konfigurationsdatei für Kibana befindet sich im Pfad $/etc/kibana/$ unter dem Namen $kibana.\ yml$. Die Struktur der Datei ist im Anhang A.4 einsehbar. Die Einstellungen hier sollten nur geändert werden, wenn bereits vorher Einstellungen an der $elasticsearch.\ yml$ vorgenommen wurden oder sich die IP-Konfiguration des Servers geändert hat. Im Betriebsumfeld, für das der ELK-Stack vorgesehen ist, werden keine weiteren Konfigurationen in der Datei benötigt.

### Logstash

Die Konfigurationsdatei für Logstash befindet sich im Pfad $/etc/logstash/$ unter dem Namen $logstash.\ yml$. Die Datei ist für den allgemeinen Betrieb aber eher irrelevant. Anpassungen am Import der Log-Dateien, Filtern und Export an Elasticsearch werden an den Dateien im Ordner $/etc/logstash/conf.\ d/$ vorgenommen. Die Konfigurationsdateien können im Anhang A.3 eingesehen werden. In der Datei $beats\ -\ input.\ conf$ kann der Port geändert werden, auf welchem Logstash die von Filebeat übertragenen Log-Dateien erwartet. Sollten weitere Log-Shipper eingesetzt werden müssen diese ebenfalls hier eingetragen werden. Verschiedene Input-Plugins finden sich in der offiziellen Elastic-Dokumentation[11]. Das Ändern oder Hinzufügen von Filtern wird in der Datei $filter.\ conf$ vorgenommen. Vor der Änderung oder dem Hinzufügen von Filtern sollte dringendst die offizielle Dokumentation von Logstash und grok konsultiert werden (Fußnote 7 und 8 auf dieser Seite). Vor der Änderung der Datei sind immer Backups der Originaldatei zu erstellen. In der Datei $elasticsearch\ -\ output.\ yml$ werden die IP-Adresse des Elasticsearch-Servers eingetragen, an den die Log-Dateien geschickt werden sollen,

[10] Grok Dokumentation, Elastic. URL: https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html (abgerufen am 10.05.2022)
[11] Logstash Input plugins, Elastic. URL: https://www.elastic.co/guide/en/logstash/current/input-plugins.html (abgerufen am 10.05.2022)

sowie der Port. Im Index-Bereich wird den übertragenen Logs noch Daten angehangen, wie z.B. ein Zeitstempel.

### Filebeat

Filebeat wird auf den Clients installiert, von denen Log-Dateien gesammelt werden sollen. Die Installation von Filebeat findet sich unter $/usr/share/filebeat$ und die Konfigurationsdateien unter $/etc/filebeat/$ in der Datei $filebeat.\ yml$. Der Aufbau der Konfigurationsdatei kann auch im Anhang unter A.6 eingesehen werden. Wichtig ist hier die korrekte Angabe der Log-Pfade sowie der IP-Adresse des Logstash-Servers. Die Struktur im Bereich Input sollte beibehalten werden. $sourcefiletype$ kann weggelassen werden, ist aber nützlich, wenn man zukünftig mit grok weitere Filterregeln anlegen möchte.

### Hardware-Informationen der Server-VM:

| | |
|---|---|
| Betriebssystem: | Debian 11 64-bit |
| Festplattenspeicher: | 1000 GiB |
| Prozessor: | 4 Sockel mit 4 CPU-Kernen |
| Arbeitsspeicher: | 32 GiB |
| Netzwerk: | DEFAULT_VLAN |
| IP-Adresse: | 192.168.10.100 |
| Subnetzmaske: | 255.255.255.0 |
| Gateway: | 192.168.0.1 |
| DNS: | 192.168.10.50 |

Bei der Installation von Debian wurde die sogenannte Minimal-Version installiert. Es wurden dementsprechend noch folgende Komponenten nachinstalliert, die für den Betrieb des ELK-Stacks nötig waren:

| **Programm** | **Version** |
|---|---|
| Gnupg2 | 2.2.35 |
| wget | 1.10 |
| cURL | 7.83.0 |

### Ordnerstruktur

Alle ELK-Stack Komponenten nutzen 2 Verzeichnispfade: Installationspfade und Konfigurationspfade. Die Konfigurationspfade bei allen 3 Komponenten sind unter $/etc/\ \%KOMPONENTE\%$ zu finden, z.B. $/etc/elasticsearch/$. Die Installationspfade befinden sich unter $/usr/share/\%KOMPONENTE\%$, also z.B. $/usr/share/elasticsearch/$.

## A.11.2 Anwenderdokumentation

# Anwenderdokumentation für die Nutzung von Kibana

### Anmeldung:

Die Weboberfläche von Kibana lässt sich über 192.168.10.100:5601 erreichen. Die Funktionalität wurde sowohl mit Mozilla Firefox als auch Google Chrome und Microsoft Edge getestet. Andere Browser sollten ebenfalls funktionieren. Sollten allerdings Probleme beim Aufruf der Seite auftreten versuchen Sie bitte zuerst einen der getesteten Browser. Die Anmeldung erfolgt mit folgenden Daten:

Zugangsdaten:

| | |
|---|---|
| Benutzername: | o********** |
| Passwort: | ********** |
| IP-Adresse: | 192.168.10.100:5601 oder o********** |

Bitte beachten Sie, dass der Dienst nur aus dem internen Firmennetzwerk zu erreichen ist. Sollten Sie sich im mobilen Arbeiten befinden stellen Sie bitte sicher, dass Sie ihre VPN-Verbindung gestartet haben und diese auch erfolgreich aufgebaut wurde.

### Allgemeine Hinweise

Nachfolgende Bilder wurden aufgrund der Größe und besseren Lesbarkeit teilweise geschnitten oder skaliert oder stellen nur Auszüge einer Oberfläche dar.

### Nutzung

Nach erfolgreicher Anmeldung wird die Start-Oberfläche angezeigt. Mit einem Klick auf den Menü-Button oben links und einem Klick auf „Discover“ gelangen Sie in die Log-Übersicht. Andere Bereiche werden zwar angezeigt, sind aber aufgrund von Nutzerbeschränkungen nicht auswählbar.

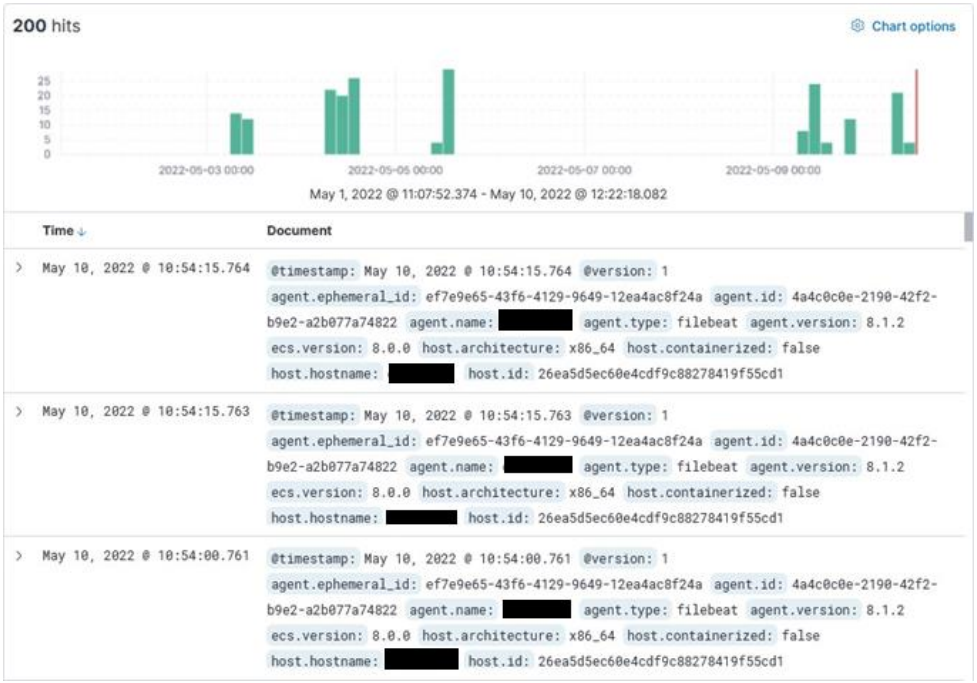



*Abbildung 1 Live-Feed in Kibana*

Hier finden sich der Live-Feed der Log-Übersicht. Standardmäßig aktualisiert Kibana diesen Feed in regelmäßigen Abständen. Der Zeitrahmen der Anzeige kann allerdings oben rechts auf der Seite justiert werden:

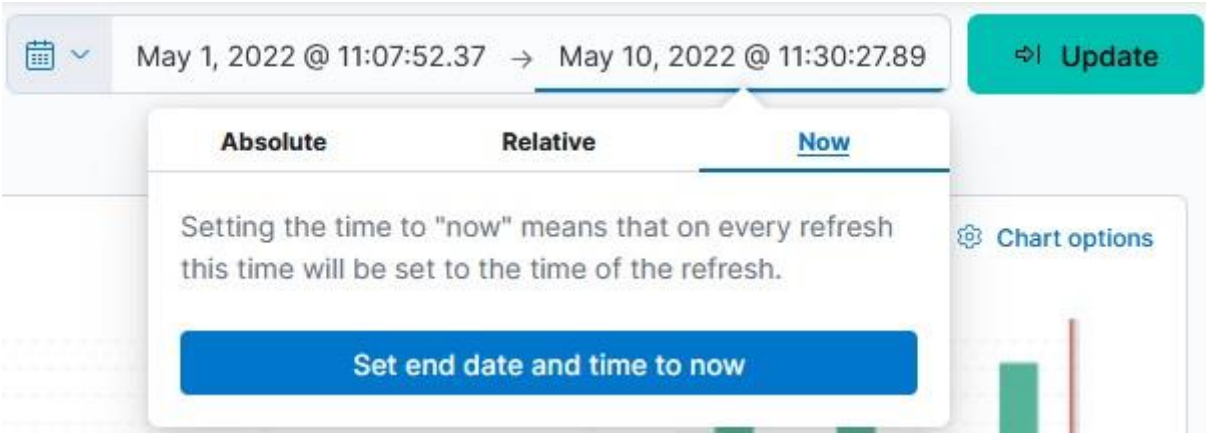



*Abbildung 2 Zeiteinstellung des Live-Feeds*

„Absolute“ lässt einen Zeitpunkt im Format Tag/Monat/Jahr festlegen, „Relative“ legt einen Zeitpunkt x Minuten vor oder nach der aktuellen Zeit fest und „Now“ setzt das System auf einen Echtzeitaktualisierungs-Modus.

Suchfilter werden über die Schaltfläche „Add Filter“ angelegt:

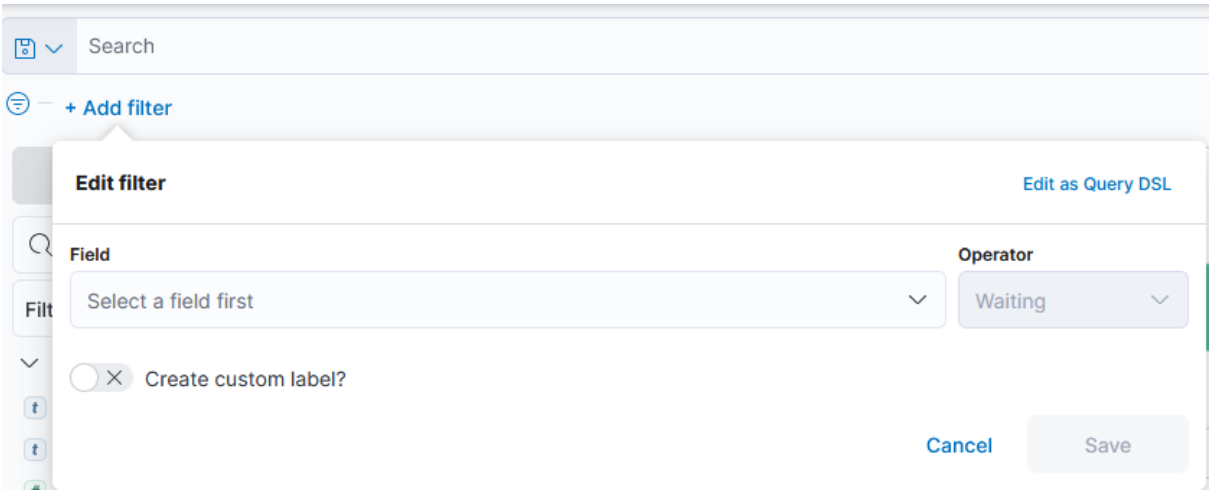



*Abbildung 3 Filter in Kibana erstellen*

Unter „Field“ muss der Name des von Logstash generierten Feldes angegeben werden. Dies sind z.B. $host.\ ip,\ host.\ hostname$ oder $host.\ mac$. Im Feld „Operator“ gibt es verschiedene Operatoren:

| | |
|---|---|
| is | Suchergebnis stimmt genau mit Suchbegriff überein |
| is not | Suchergebnisse beinhalten nicht den Suchbegriff |
| is one of | Suchergebnis beinhaltet einen von verschiedenen Suchbegriffen |
| is not one of | Suchergebnis beinhaltet keinen von verschiedenen Suchbegriffen |
| exists | Listet alle Ergebnisse, die für das „Field“ einen Wert haben |
| does not exit | Listet nur Ergebnisse, bei denen der Wert „Field“ leer ist |

*Tabelle 11 Operatoren in Kibana*

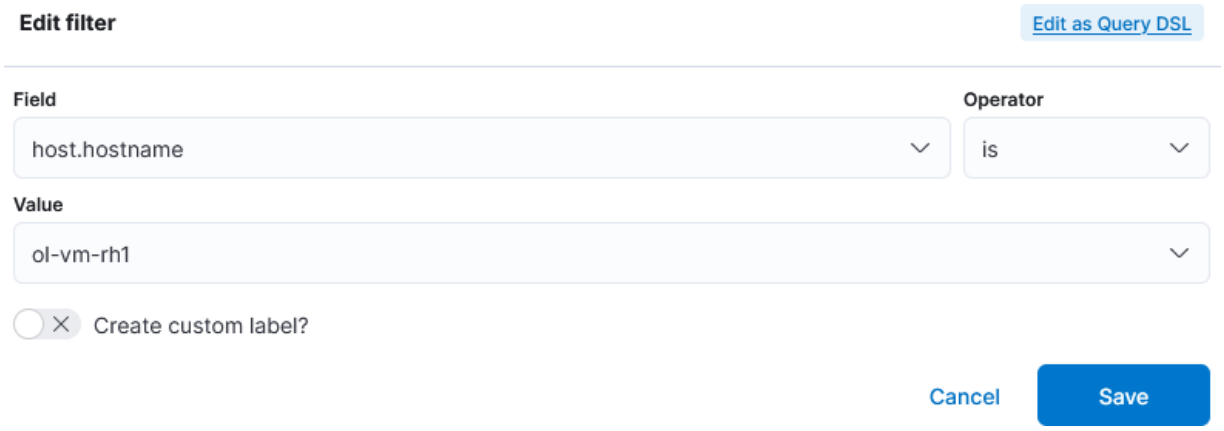



*Abbildung 4 Beispielfilter erstellen*

Stellt man den Filter beispielweise wie in Abbildung 4 ein, werden nur Log-Ereignisse angezeigt, die von dem Client [redacted] stammen. Zusätzlich werden alle Instanzen des Suchbegriffes farblich hervorgehoben.


*Abbildung 5 Log-Ereignisse mit angewandtem Filter*

In der linken Hälfte der grafischen Oberfläche gibt es verschiedene Felder, nach denen der Live-Feed geordnet werden kann. Durch einen Klick auf das blaue Plus-Zeichen werden diese Filter angewendet. Das blaue Plus-Zeichen taucht erst auf, wenn mit der Maus über den entsprechenden Eintrag gefahren wird:

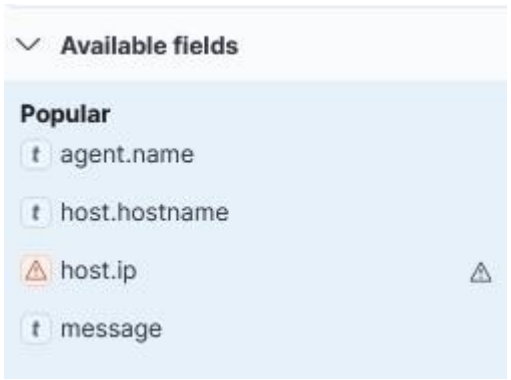



*Abbildung 6 Feld-Filter*

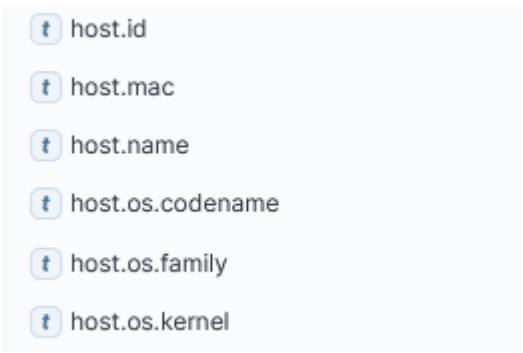



*Abbildung 7 Filter-Auswahl*

Damit lassen sich eigene Suchoberflächen zusammenstellen, um bereits vorab nach relevanten Eigenschaften zu suchen. Eine Auswahl der folgenden Feld-Filter führt zu einer neuen Live-Feed Ansicht (siehe Abbildung 8).


*Abbildung 8 angepasster Live-Feed mit Hilfe von Filtern*

Da die eigentlichen Log-Daten im „Message“ Feld gespeichert werden kann man über die Filterfunktion auch direkt nach Werten der Log-Daten suchen, z.B. Kundenauftragsnummern oder Rechnungsnummern.

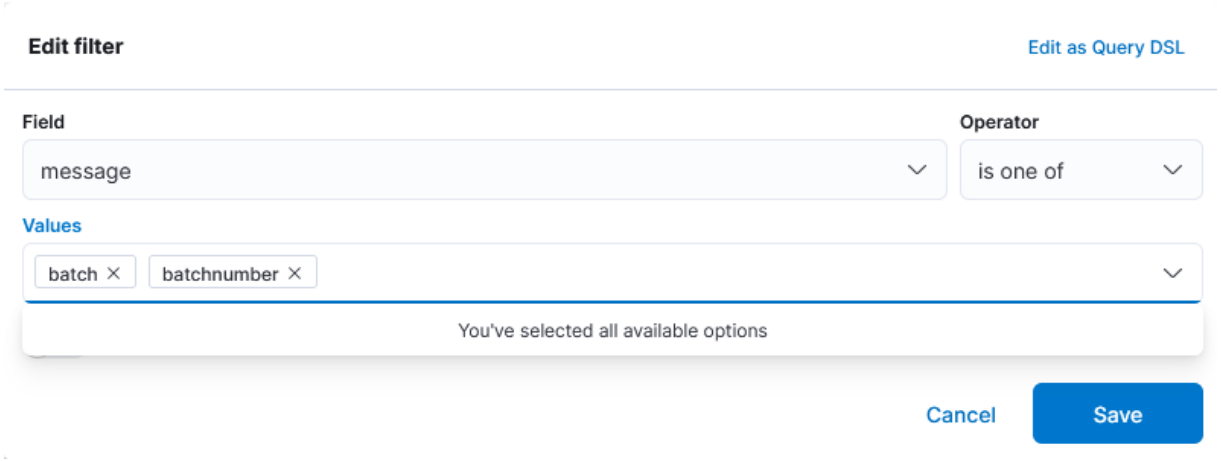



*Abbildung 9 Log-Daten nach Keywords durchsuchen*


*Abbildung 10 Ergebnisse mit einem/mehreren Keywords*

## Abschluss

Kibana ist ein mächtiges Werkzeug für erfahrene Anwender und beschleunigt die Suche nach Fehlern oder Auffälligkeiten enorm. Die in dieser Dokumentation gezeigten Befehle und Beispiele sind nur ein kleiner Teil der Möglichkeiten, die Kibana bietet.

## Ansprechpartner

Bei Fragen, Problemen oder Verbesserungsvorschlägen wenden Sie sich bitte an eine der folgenden Personen:

| **Name** | **E-Mail** | **Telefon** |
|---|---|---|
| Kai Müller | XXXXXXXXXX | XXXXXXXXXX |
| Klaus Tornow-Frerichs | XXXXXXXXXX | XXXXXXXXXX |

*Tabelle 12 Kontaktdaten der Verantwortlichen*

## A.12 Amortisierung des Projektes

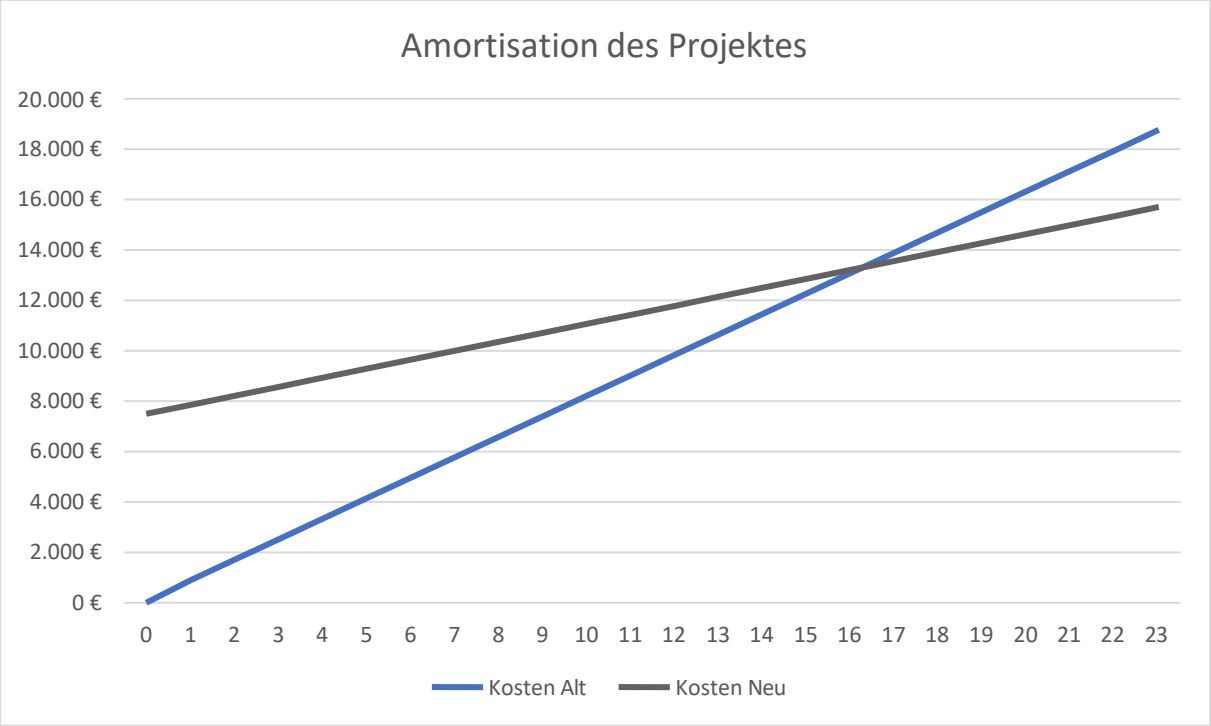



*Abbildung 11 Amortisierung*

| Schritt | Zeit (geplant) | Zeit (tatsächlich) |
|---|---|---|
| **Planungsphase** | **11** | **8** |
| Kundengespräch | 2 | 1 |
| IST-Analyse | 2 | 0,5 |
| SOLL-Konzept | 2 | 1 |
| Besprechung des Konzeptes mit Kunden | 2 | 1 |
| Softwarevergleich | 0,5 | 1 |
| Nutzwertanalyse | 0,5 | 1 |
| Angebot für Server einholen | 1 | 1 |
| Vergleich Hardware und VM | 1 | 1 |
| Datenschutz | 0 | 0,5 |
| **Durchführung** | **16** | **20** |
| Beschaffung von Hard- und Software | 2 | 3 |
| Erstellen der benötigten virtuellen Maschine | 1 | 2 |
| System überprüfen | 0 | 1 |
| Installation des Elastic-Stacks | 5 | 4 |
| Konfiguration des Elastic-Stacks | 3 | 4 |
| Start des ELK-Stacks | 1 | 1 |
| Installation von Filebeat auf Clients | 2 | 2 |
| Konfiguration von Filebeat | 2 | 3 |
| **Abschlussphase** | **8** | **7** |
| Testen der Datenübertragung | 1 | 1 |
| Anfertigen einer Benutzerdokumentation | 1 | 1 |
| Anfertigen einer Administratordokumentation | 1,5 | 1 |
| Kostenkalkulation | 0,5 | 1 |
| Übergabe an den Kunden | 2 | 2 |
| Abschlussgespräch | 2 | 1 |
| **Gesamt** | **35** | **35** |

*Tabelle 13 Zeitplanung des Projektes*

## A.13 Eidesstattliche Erklärung

Entfernt für