Abschlussprüfung Sommer 2020

Fachinformatiker für Systemintegration

Dokumentation zur betrieblichen Projektarbeit

Evaluierung von „Icinga 2“

Open Source Monitoring

Abgabetermin: Augsburg, den 21.05.2020

Prüfungsbewerber:
Andreas Germer
Hagelbach 9
86316 Bachern

Ausbildungsbetrieb:
KUKA AG
Zugspitzstraße 144
86163 Augsburg

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

UEFI	Unified Extensible Firmware Interface; Nachfolger von BIOS
BIOS	Basic Input/Output System; Firmware eines Computers
PHP	Hypertext Preprocessor; Skriptsprache zur Erstellung von Websites
CPU	Central Processing Unit; Hauptprozessor
RAM	Random-Access Memory; Arbeitsspeicher
VM	Virtuelle Maschine
LAN	Local Area Network
IDO	Icinga Data Output
LDAP	Lightweight Directory Access Protocol; Protokoll zur Abfrage von Verzeichnisdiensten
FQDN	Full Qualified Domain Name; eindeutige Adresse eines Hosts im Netzwerk
HTTP	Hypertext Transfer Protocol; Protokoll zur Übertragung von z.B. Websites
URL	Uniform Resource Locator; hier: Internetadresse
ESXi	Typ-1 Hypervisor von VMware
ISO	Speicherabbild eines Dateisystems

1 Einleitung

1.1 Projektumfeld

Die KUKA AG ist ein international tätiges Unternehmen in der Maschinenbau- und Automatisierungsbranche mit rund 14.200 Mitarbeitenden. Zum Produktportfolio zählen neben Industrierobotern auch die Planung und der Bau von Produktionsstraßen.

Die IT-Infrastruktur am Standort Augsburg besteht aus ca. 1.200 größtenteils virtualisierten Servern; an anderen Standorten befindet sich vereinzelt eine kleinere Anzahl an Servern. Es kommen alle gängigen Betriebssysteme zum Einsatz. Dieses Projekt wurde durch die Abteilung „Datacenter & Network“, die den Betrieb der Rechenzentren sowie des internen IT-Netzwerks der KUKA AG überwacht und koordiniert, in Auftrag gegeben.

1.2 Projektziel

Momentan wird kein einheitliches Monitoring der IT-Systeme durchgeführt. Die verschiedenen Standorte und Abteilungen setzen auf unterschiedliche und teils veraltete Lösungen; zur besseren Administration sollen diese durch ein einheitliches und zentrales System ersetzt werden.

Es soll evaluiert werden, ob die freie Monitoringsoftware „Icinga 2“ für den firmeninternen Einsatz geeignet ist. Dazu werden in einem ersten Schritt Anforderungen der Process Owner und Systemadministrierenden gesammelt. Anschließend wird eine Testumgebung eingerichtet, um zu prüfen, inwieweit die Ansprüche des Unternehmens durch „Icinga 2“ erfüllt werden können. Abschließend werden die Ergebnisse, auch in Hinsicht auf betrieblichen Vorgaben wie die der IT-Sicherheit, analysiert und eine Empfehlung ausgesprochen.

1.3 Projektbegründung

Durch die zunehmende Digitalisierung aller Geschäftsprozesse sind Unternehmen äußerst abhängig von Computersystemen. Wird das Schutzziel der Verfügbarkeit nicht ausreichend gut verfolgt, kommt es zu Systemausfällen und der Geschäftsbetrieb ist nicht länger möglich - mit unabsehbaren wirtschaftlichen Folgen. Um Ausfallsicherheit zu gewährleisten, ist ein umfangreiches und zuverlässiges Monitoring aller Ressourcen unerlässlich.

1.4 Projektabgrenzung

Als Monitoringsystem interagiert „Icinga 2“ potenziell mit allen Geräten und Diensten im Netzwerk. Für dieses Projekt wird das Zusammenspiel auf virtuelle Maschinen mit ausgewählten, geeigneten Betriebssystemen und Diensten eingegrenzt.

2 Projektplanung

2.1 Projektphasen

Das Projekt wird innerhalb einer 35-Stunden-Arbeitswoche durchgeführt. Die einzelnen Phasen überschneiden sich teils; beispielsweise können virtuelle Maschinen bereits eingerichtet werden, während noch auf Rückmeldungen bezüglich der Anforderungen an das System gewartet wird.

Planungsphase		Durchführungsphase		Projektabschluss	
Ist-Analyse	*2h*	Einrichtung Wirt und VMs	*6h*	Ergebnisbewertung	*2h*
Soll-Analyse	*2h*	Installation Icinga	*2h*	Empfehlungsformulierung	*1h*
Definierung Kriterienkatalog	*3h*	Installation Testsysteme	*3h*	Dokumentation	*7h*
		Testdurchführung	*7h*		

Tabelle 1: Zeitplanung

2.2 Abweichungen vom Projektantrag

Es zeigte sich im Vorfeld, dass ein klassischer Soll-Ist-Vergleich für dieses Projekt nicht sinnvoll ist, da das Hauptaugenmerk auf der Evaluation der Lösung liegt. Die vorhergesehene Zeit wurde stattdessen zur Ergebnisbewertung eingeplant.

2.3 Ressourcenplanung

Folgende Hardware wird zur Durchführung des Projekts benötigt:

- Notebook mit RJ-45 Port und USB Port
- Workstation mit mindestens: 2x RJ-45 Port, 2x USB Port, 16 GB Arbeitsspeicher, 500 GB HDD
- Software: Windows 10 (min. Professional), Windows Server (min. 2012), Debian (aktuelle Version), Ubuntu (aktuelle Version), ESXi (aktuelle Version), aktueller Webbrowser, SSH-Client
- 2x Netzwerkkabel, min. 1,5m
- USB-Datenträger, 8 GB

3 Planungsphase

3.1 Ist-Analyse

Zu Projektbeginn existiert im Unternehmen keine einheitliche Lösung zur Serverüberwachung. Für den Großteil der Server am Standort Augsburg wird der „Advanced Host Monitor“ eingesetzt. Diese Lösung erhält trotz ihres Alters noch regelmäßige Updates. Aufgrund eines veralteten User Interfaces, einem beschränktem Funktionsumfang und der nicht mehr ausreichenden Leistungsfähigkeit wird jedoch seit vielen Jahren der Wunsch nach einem neuen Monitoring-System geäußert.

Die Entwicklungsabteilungen überwachen die von ihnen betreuten Server mit einer auf „Elasticsearch“ und „Kibana“ basierenden Lösung. Um Know-how zu bündeln und Personalkosten zu sparen, signalisierten die Verantwortlichen eine Bereitschaft zur Zusammenlegung des Servermonitoring.

Am Standort Bremen wird seit mehreren Jahren auf die Open-Source-Anwendung „Icinga“ gesetzt. Die Systembetreuenden haben mit dieser Lösung positive Erfahrungen gesammelt, und loben insbesondere die einfache Erweiterbarkeit durch Plugins sowie die Möglichkeit ohne Aufwand optisch ansprechende Dashboards zu erstellen. Aufgrund der Komplexität dieser Software und einem anstehenden Update auf „Icinga 2“ wurde auch von dieser Seite der Wunsch nach einem einheitlichen und zentral verwaltetem Servermonitoring geäußert.

3.2 Soll-Analyse

3.2.1 Befragung der Fachabteilungen

Es wurden die entsprechenden Verantwortlichen via E-Mail, Telefonaten und Meetings zu den betreuten Servern und deren Monitoring befragt. Hierbei zeigten sich große Überschneidungen bei den Serverumgebungen und den Anforderungen an deren Überwachung, was eine Zusammenlegung logisch erscheinen lässt.

Über alle Abteilungen hinweg werden verschiedenste Windows-Versionen sowie Linux-Distributionen eingesetzt, weswegen ausschließlich ein plattformunabhängiges Monitoring-System wie „Icinga 2“ eingesetzt werden kann. In Abstimmung mit den Fachabteilungen wurden drei zu testende Betriebssysteme ausgewählt (siehe 4.1.3 Installation der virtuellen Maschinen). Die zu überwachenden Parameter beziehungsweise Dienste ähneln sich zwischen den Abteilungen ebenfalls sehr stark: Neben Leistungsmetriken wie Prozessorauslastung, Arbeitsspeicherbelegung oder freiem Massenspeicher wird viel Wert auf die Überprüfung der Erreichbarkeit im Netzwerk gelegt. Aufgrund der zunehmenden Verbreitung von webbasierten Anwendungen soll auch der Betrieb von Webservern (plattformunabhängig) überwacht werden. Weiterhin wurde der Wunsch nach einer ansprechenden grafischen Aufbereitung in einer Weboberfläche geäußert.

3.2.2 Kriterienkatalog

Auf Basis der Befragungen wurde folgender Kriterienkatalog aufgestellt:

- Überwachbare **Betriebssysteme**: Alle gängigen Windows Server Versionen und Linux-Distributionen
- Überwachbare **Systemparameter**: CPU-Auslastung, RAM-Belegung, Massenspeicherbelegung
- Überwachung von **Webservern**
- Ansprechende und anpassbare **grafische Aufbereitung** im Webbrowser

3.3 Wirtschaftlichkeitsanalyse

3.3.1 Projektkosten

Die Kosten, die durch das Projekt verursacht werden, setzen sich sowohl aus Personal- als auch aus Ressourcenkosten zusammen. Die Berechnung der (fiktiven) Stundensätze findet sich im Anhang A.1: Berechnung des Stundensatzes von Mitarbeitenden auf Seite i.

Die Kosten für die 35-Stunden-Woche des Auszubildenden belaufen sich auf 350 €. Die Arbeitszeit, die bei mitarbeitenden Personen zur Durchführung des Projekts angefallen ist (z.B. für Befragungen, Konfiguration der Netzwerkkomponenten, Anpassungen der Firewall für Updates, Abnahme) wurde auf vier Stunden geschätzt. Dafür fielen Personalkosten in Höhe von 160 € an. Die für das Projekt eingesetzte Hardware ist bereits abgeschrieben, und wird mit einer Pauschale von 360 € pro Jahr verrechnet. Auf die Projektdauer von fünf Tagen ergibt das Betriebskosten von 4,93 €.

Eine Aufstellung der Projektosten befindet sich in Tabelle 2. Sie betragen insgesamt 514,93 €.

Vorgang	Zeit	Kosten	Gesamtkosten
Projektdurchführung	35 h	10 € / h	350,00 €
Kollegiale Unterstützung	4 h	40 € / h	160,00 €
Betriebskosten Server	5 d	360 € / a	4,93 €
			514,93 €

Tabelle 2: Kostenaufstellung Projekt

3.3.2 Betriebskosten des Monitorings

Die Betriebskosten für ein Monitoringsystem umfassen Personalkosten für Wartungsarbeiten und Anpassungen, sowie die Kosten die für zwei redundant ausgelegte Hardware-Server anfallen. Diese werden benötigt, da das Monitoring nicht innerhalb der Virtualisierungsumgebung betrieben werden sollte, um eine funktionierende Serverüberwachung auch bei Ausfall der VM-Infrastruktur zu gewährleisten.

Für das Einpflegen neuer Server oder Funktionen wird eine Arbeitsstunde pro Woche veranschlagt. Weiterhin wird der Arbeitsaufwand für Updates und Fehlerbehebungen nach Erfahrungsberichten auf acht Stunden pro Quartal geschätzt. Auf ein Jahr summiert ergibt beides einen Arbeitsaufwand von insgesamt 84 Stunden.

Die geschätzten Kosten für einen Hardware-Server belaufen sich auf 4.000 € jährlich. Die Gesamtkosten für den Betrieb einer Monitoring-Instanz belaufen sich, wie in Tabelle 3 dargelegt, auf 11.360,00 € pro Jahr.

Beschreibung	**Einzelkosten**	**Einheit**	**Jahreskosten**
Wartungs- und Pflegearbeiten	40,00 € / h	84 h / a	3.360,00 €
Hardwarekosten	4.000,00 € / a	2	8.000,00 €
			11.360,00 €

Tabelle 3: Kostenaufstellung Monitoring

3.3.3 Amortisationsdauer

Sollte die Evaluation durch diese Projektarbeit ergeben, dass „Icinga 2“ für den firmenweiten Einsatz geeignet ist, werden drei momentan im Betrieb befindliche Monitoringinstanzen (siehe 3.1 Ist-Analyse) durch ein zentrales „Icinga 2“-System ersetzt (siehe 1.2 Projektziel). Die Betriebskosten für die derzeitigen Überwachungssysteme entsprechen etwa den in 3.3.2 Betriebskosten des Monitorings berechneten 11.360,00 € pro Jahr; die jährliche Einsparung, wenn statt drei Systemen nur noch eines betrieben wird, beläuft sich auf:

$$11.360{,}00\,€ \cdot 2 = 22.720{,}00\,€ \tag{1}$$

Für die Ersteinrichtung des neuen Systems werden etwa zwei Wochen benötigt, also insgesamt 70 Arbeitsstunden. Das verursacht einmalige Kosten in Höhe von:

$$70\text{h} \cdot 40{,}00\,€/\text{h} = 2.800\,€ \tag{2}$$

Die Amortisationszeit beträgt somit:

$$\frac{2.800\,€}{22.720{,}00\,€/\text{a}} \approx 0,123 \text{ Jahre} \approx 45 \text{ Tage} \tag{3}$$

4 Durchführungsphase

4.1 Vorbereitung der Tests

4.1.1 Vorbereiten der Hardware

Eine ausrangierte DELL Precision Workstation dient als Hardwareplattform für das Projekt. Die Ausstattung von 32 Gigabyte Arbeitsspeicher und einem Intel Xeon E5-1650 v2 Hochleistungsprozessor ist ausreichend für den Betrieb von mehreren virtuellen Maschinen.

Um den Bedingungen in einem „echten“ Rechencenter möglichst nahe zu kommen, wird der VM-Hypervisor mit zwei Netzwerkschnittstellen ausgestattet. Ein Interface ist für die Kommunikation mit dem Hypervisor selbst vorgesehen (sogenanntes Management-LAN), und eine weitere Netzwerkschnittstelle wird von den virtuellen Maschinen benutzt, um im Netzwerk erreichbar zu sein. Da die Workstation nur über einen Netzwekport verfügte, musste eine zweite Netzwerkkarte eingebaut werden.

4.1.2 Installation des Hypervisors

Um die gegeben Ressourcen optimal auszunutzen, wird ein sogenannter Typ 1-Hypervisor eingesetzt. Dieser kommuniziert direkt mit der Hardware, ohne dass dazwischen ein anderes Betriebssystem zum Einsatz kommt. Für dieses Projekt wird das System „ESXi“ von VMware in der Version 6.7 verwendet, das auch in den firmeneigenen Rechenzentren zum Einsatz kommt.

Die Installation gestaltet sich als simpel. Nachdem wichtige UEFI-Optionen angepasst wurden (UEFI statt Legacy-BIOS; hardwareseitige Virtualisierungsunterstützung) wird das System von einem USB-Datenträger, der zuvor mit dem ESXi-Image geflasht wurde, gestartet. Nachdem die für die Installation zu benutzende Festplatte ausgewählt, und ein Root-Passwort vergeben wurde, startet der Installationsvorgang. Nach abgeschlossener Installation muss noch eine IP-Adresse für das Managementnetzwerk vergeben werden.

Die restliche Konfiguration geschieht komfortabel über eine Weboberfläche. Der Lizenzschlüssel wird hinterlegt, die zweite Festplatte als Datastore eingebunden, und das Netzwerk für die virtuellen Maschinen konfiguriert (siehe Abbildung 1). Hierzu wird ein neuer virtueller Switch für den zweiten Netzwerkport (der, der nicht mit dem Management-LAN belegt ist) erstellt und mit einer neuen Port Group versehen. Diese Port Groups werden in ESXi verwendet, um logische Netzwerkschnittstellen bereit zu stellen und zu verwalten.

4.1.3 Installation der virtuellen Maschinen

Für virtuelle Maschinen muss in der ESXi-Weboberfläche zunächst die Systemkonfiguration festgelegt werden. Hierfür werden die für die VM vorgesehene CPU-Kerne, Arbeits- und Massenspeicherspeicherkapazität (siehe Abbildung 2) sowie die zu benutzenden Port Groups eingestellt. Abschließend wird in ein virtuelles DVD-Laufwerk das ISO-Abbild für das entsprechende zu installierende Betriebssystem eingehängt, und die virtuelle Maschine gestartet. Die anschließende Betriebssysteminstallation kann über eine emulierte Browser-Konsole durchgeführt werden.

Um in diesem Testsystem möglichst nah an das Firmennetzwerk heranzukommen, wurden drei Betriebssysteme für die Server ausgewählt: Windows Server 2019, Ubuntu 18.04 LTS sowie Debian 10.3. Die Installation der Systeme verlief ohne Probleme.

4.1.4 Installation von „Icinga“

Die Installation von „Icinga“ unter Ubuntu beginnt mit dem Aufruf der Rootshell und einem anschließendem kompletten Systemupdate. Die Pakete `icinga2` und `icingacli` sind für die Monitoring-Engine an sich, sowie die Verwaltung über Kommandozeile zuständig und können über die Paketverwaltung (hier: apt) installiert werden. Das Paket `monitoring-plugins` beinhaltet Plugins für die wichtigsten Monitoringaufgaben (z.B. Ping). [1]

Aufgrund der firmeninteren Verbreitung, sowie der In-Memory Unterstützung, wird MySQL als Datenbanksystem verwendet. Durch Installation der Pakete `mysql-server` und `mysql-client` werden ein MySQL-Server sowie ein MySQL-Client bereitgestellt, das Skript `mysql_secure_installation` (siehe Abbildung 3) verbessert die Sicherheit durch Maßnahmen wie das Entfernen von anonymen Accounts oder die Absicherung des root-Accounts.

Im nächsten Schritt muss die Schnittstelle zwischen „Icinga Data Output“ (Exportfunktion für Monitoringdaten; kurz IDO) und MySQL geschaffen werden, damit die gesammelten Daten auch gespeichert werden können. Dies geschieht durch Installation des Pakets `icinga2-ido-mysql`. Anschließend kann das Webinterface `icingaweb2` installiert werden.

Beim Versuch, das installierte Web-Frontend aufzurufen, wurde lediglich ein PHP-Fehler angezeigt. Es ergab sich, dass dieser durch Inkompatibilitäten mit bestimmten PHP 7.2 Plugins verursacht wird. Dieser Fehler wurde bereits Mitte 2018 behoben, allerdings ist der Bugfix anscheinend nicht in das Paket eingespielt worden. Das Problem wurde gelöst, indem die aktuelle Version von „Icinga Web 2“ vom öffentlichen Github-Repository nach `/usr/share/icingaweb2` geklont wurde.

Ist „Icinga Web 2“ fertig installiert, muss noch ein Benutzer für die MySQL-Datenbank erstellt werden. Abschließend wird IDO mittels `icinga2 feature enable command ido-mysql` aktiviert, und ein Setup-Token mit `icingacli setup token create` generiert. Nach einem Neustart des icinga2-Dienstes (`systemctl restart icinga2`) ist „Icinga 2“ bereit, eingerichtet zu werden.

[1] Eine komplette (nachträglich optimierte) Liste an Befehlen, die für eine komplette Installation unter Ubuntu 18.04 ausgeführt werden muss, findet sich im Anhang A.2: Installationsskript für Ubuntu 18.04 auf Seite ii

Einrichtung über Webfrontend Im Browser wird nun (IP-Adresse des Servers)/icingaweb2/setup aufgerufen, um den Konfigurationsassistenten (siehe Abbildung 5) zu starten. Nach Eingabe des vorher generierten Setup-Tokens wird zunächst geprüft, ob alle notwendigen PHP-Plugins vorhanden sind. In der für diese Projekt durchgeführten Installation fehlte das Plugin „PDO-PostgreSQL“, welches aber aufgrund der Verwendung von MySQL nicht benötigt wird, sowie „cURL“, das mit der Paketverwaltung nachinstalliert wurde.

Im Abschnitt „Configuration“ werden noch weitere kleinere Einstellungen getroffen, hierbei bietet sich aber an, die Standardeinstellungen beizubehalten. Eine Ausnahme stellt die bevorzugte Authentifizierungsmethode für Nutzende des Frontends dar; hier kann zwischen LDAP und einer MySQL-Datenbank gewählt werden. Für dieses Projekt wurde aufgrund des geringeren Umfangs letztere Option gewählt, die Konfiguration hierfür findet sich in Abbildung 6.

Einpflegen von Servern Durch Installation des Plugins „Icinga Director“ kann das Hinzufügen von zu überwachenden Servern über die Webschnittstelle erledigt werden; für dieses Projekt wird darauf verzichtet und die weitergehende Konfiguration geschieht über Anpassung der Konfigurationsdateien unter `/etc/icinga2/conf.d/`. Dort können in der Datei `hosts.conf` neue Server hinzugefügt werden (siehe Abbildung 8). Abschließend kann die Syntax der Konfigurationsdateien mittels `icinga2 daemon -C` überprüft werden; nach einem Neustart des „Icinga 2“ Dienstes sind die neuen Hosts im Monitoring verfügbar.

Installation des Agents Mit der bisherigen Konfiguration können einfache Checks (z.B. Ping) auf die entsprechenden Hosts durchgeführt werden. Für aufwendigere Überprüfungen wie Speicherplatzkontrolle muss auf den Hosts eine Software installiert werden. Hierfür muss zunächst der Monitoringserver als „Master“ deklariert werden, dies geschieht mittels des Befehls `icinga2 node wizard`. Anschließend wird mit `sudo icinga2 pki ticket -cn (FQDN)` noch ein „Ticket“ für den angegebenen FQDN erstellt, mithilfe dessen der Host die Verbindung zum Master herstellen kann.

Der Host selbst installiert die Software ebenfalls. Auf Linux-basierten Systemen ist diese im Paket `icinga2` enthalten, für Windows existieren herunterladbare Installationsdateien. Mit Ausführung des Befehls `icinga2 node wizard` wird die Konfiguration des Agents abgeschlossen, bei Windows geschieht dies über ein grafisches Userinterface. Nachdem die Adresse des Masters und die eben erstellte Ticketnummer angegeben wurden, stellt der Agent eine Verbindung zum Monitoring her.

4.2 Durchführung der Tests

4.2.1 Überwachung von Leistungsparametern

Das installierte Pluginpaket `monitoring-plugins` beinhaltet Plugins zur Überwachung der Leistungsparametern (CPU-Auslastung, RAM-Belegung und belegter Massenspeicher). Für diese Überprüfungen müssen in `/etc/icinga2/conf.d/commands.conf` Definitionen erstellt werden. Am Beispiel in Abbildung 9 ist erkennbar, dass diese Definitionen neben dem auszuführendem Befehl (diese sind

als Bash-Skript abgelegt; das Verzeichnis hierfür ist in der Konstante „PluginDir“ gespeichert) auch die zu übergebenden Argumente enthält. Im Beispiel der CPU-Auslastung wird mitgegeben, dass der Durchschnitt aller CPU-Kerne berechnet (`-r`) und bei bestimmten Schwellenwerten (`--critical`) ein Alarm gegeben werden soll.

Ist die Definition erfolgreich validiert, muss sie noch den entsprechenden Hosts zugewiesen werden. In der Datei `/etc/icinga2/conf.d/services.conf` wird hierzu ein Eintrag, wie in Abbildung 10 gezeigt, erstellt; dieser gibt den entsprechenden Befehl (`check_command`) und die zu überprüfende Instanz (`command_endpoint`) an und weist den Befehl zum Schluss allen Hosts mit einer Adresse zu (`assign where host.address`).

Die Konfiguration der anderen Checks (RAM-Auslastung und Massenspeicherbelegung) läuft gleichermaßen. Die Überwachung der nach dem Kriterienkatalog benötigten Parameter funktioniert problemlos und die Werte decken sich mit den von den Systemen ermittelten Auslastungen. Somit ist „Icinga 2“ hierfür einsatzfähig.

4.2.2 Betriebssystemkompatibilität

Es konnten keine Probleme bei der Integration von anderen Betriebsystemen festgestellt werden. Sowohl unter Ubuntu 18.04, Debian 10 und Windows Server 1809 läuft „Icinga 2“ ohne Probleme. Eine umfassende Betriebssystemkompatibilität ist somit gewährleistet.

4.2.3 Webserverüberwachung

Die Überwachung von Webservern muss analog zu den vorhergehenden Checks eingerichtet werden. Das Plugin „http“ übernimmt diese Aufgabe. In der Abbildung 11 ist gut zu erkennen, wie die Antwort der HTTP-Abfrage (Statuscode 200 OK) ausgewertet und dargestellt wird. Für die Antwortzeit lassen sich auch hier Schwellenwerte konfigurieren, sodass nicht nur Erreichbarkeit, sondern auch Performance überprüft werden kann. HTTPS-Abfragen werden durch ein gleichnamiges Plugin ebenfalls unterstützt. Die Überwachung von Webdiensten ist somit nach den in der Planungsphase aufgestellten Kriterien möglich.

4.2.4 Grafische Aufbereitung

Auf der Startseite lassen sich mehrere Dashboards erstellen, die mithilfe sogenannter „Dashlets“ (URL jedes aufrubaren Menüs) befüllt werden können. Somit ist es einfach, sehr angepasste und frei konfigurierbare Übersichten zu erstellen (siehe Abbildung 12).

Sollte dies nicht ausreichen, bietet die Entwicklercommunity viele weitere Plugins für diesen Zweck an. `dashin-dashboard` wird schon am Standort Bremen eingesetzt. Nach Auskunft der Mitarbeitenden ist dieses in der Lage, noch ansprechendere Dashboards zu erstellen. Auch hier konnten somit die vorher definierten Kriterien erfüllt werden.

5 Projektabschluss

5.1 Evaluationsergebnisse

„Icinga 2“ konnte alle definierten Anforderungen erfüllen. Die einfache Erweiterbarkeit durch Plugins, sowie die Verfügbarkeit vieler verschiedener Erweiterungen, bedingen die enorme Flexibilität dieser Monitoringlösung. Das wichtigste Kriterium, die Überwachung der Systemparameter, ist dadurch in hohem Maße erfüllt. Die Administration wirkt zwar anfangs kompliziert, durch Anpassung der Konfigurationsdateien können Routineaufgaben wie das Hinzufügen neuer Server aber leicht automatisiert werden. Da bereits Arbeitskräfte im Unternehmen mit der Vorgängerversion gearbeitet haben, fällt die Komplexität des Systems weniger ins Gewicht.

Die Software erfüllt alle betrieblichen Vorgaben bezüglich Datenschutz und IT-Sicherheit. Es werden keine personenbezogenen Daten erhoben, und die ermittelten Metriken werden mittels TLS verschlüsselt an den Monitoringserver übertragen.[2]

Weiterhin sprechen auch die allgemeinen Vorteile einer Open-Source Lösung für „Icinga 2“, wie entfallende Lizenzkosten oder die erhöhte Sicherheit. Freie Software kann außerdem länger in Unternehmen eingesetzt werden, da die Abhängigkeit zu einem bestimmten Hersteller für Sicherheitsupdates oder Ähnlichem entfällt. Es wird an dieser Stelle also eine klare Empfehlung für den Einsatz von „Icinga 2“ innerhalb der KUKA ausgesprochen; das Projektziel (siehe 1.2 Projektziel) wurde erreicht.

5.2 Abnahme

Das Projekt wurde im Rahmen einer Videokonferenz an den Projektbeauftragten präsentiert und übergeben.

5.3 Lessons Learned

Das in 4.1.4 Installation von „Icinga“ beschriebene Fehlerbild, welches nur durch Klonen des aktuellen GitHub-Archivs beseitigt werden konnte, kostete viel Zeit. Zukünftig sollten die Versionsnummern von heruntergeladener Software auf ihre Aktualität überprüft werden, um derartigen Fehlern vorzubeugen.

5.4 Ausblick

Es folgt auf die Überprüfung des Systems durch die Informationssicherheits- und Datenschutzbeauftragten zunächst ein abteilungsinterner Rollout von „Icinga 2“. Sollte sich die Lösung auch in der Praxis als tauglich herausstellen, werden nach und nach die restlichen Abteilungen an das System angebunden.

[2] Vgl. ICINGA.COM

Quellenverzeichnis

Dr. Peter Hoberg

Dr. Peter Hoberg: *Vollständige Ermittlung von Personalkosten.* https://www.controllingportal.de/Fachinfo/Kostenrechnung/Vollstaendige-Ermittlung-von-Personalkosten.html, Abruf: 03.04.2020

icinga.com

icinga.com: *Icinga 2 Technical Concepts - Communication.* https://icinga.com/docs/icinga2/latest/doc/19-technical-concepts/#communication, Abruf: 03.04.2020

igmetall-bayern.de

igmetall-bayern.de: *Tarifinfos Metall- und Elektroindustrie Bayern.* https://www.igmetall-bayern.de/metall-elektro/, Abruf: 03.04.2020

igmetall.de

igmetall.de: *Metall- und Elektroindustrie ERA – Ausbildungsvergütungen.* https://www.igmetall.de/download/docs_MuE_ERA_Ausbildung_Juni2018_9bdc083c9c0ed885c63bd1b076830a817eaec814.pdf, Abruf: 03.04.2020

A Anhang

A.1 Berechnung des Stundensatzes von Mitarbeitenden

Nach der Entgeldtabelle der IG Metall Bayern für Metall und Elektro[3] erhalten Angestellte der Entgeldgruppe EG 08 ein Monatsbruttogehalt von 3.800,00 €. Zusätzlich dazu müssen ungefähr ein Fünftel für die Arbeitgeberanteile der Sozialversicherung und Mehrleistungen wie Urlaubsgeld hinzugerechnet werden.[4] Aufsummiert ergibt das jährliche Personalkosten von 54.720,00 €.

$$3.800{,}00\,€ \cdot 1{,}2 \cdot 12\ \mathrm{m} = 54.720{,}00\,€\ /\mathrm{a} \tag{1}$$

Eine Arbeitszeit von 35 Stunden pro Woche bedeuetet bei 52 Wochen eine Jahresarbeitszeit von 1820 Stunden. Abzüglich der Urlaubstage (30), Feiertage (ca. 11) und Fehltage durch Krankheit und Fortbildungen (ca. 15) ergibt das eine Jahresarbeitszeit von 1498 Stunden pro Jahr.

$$35\mathrm{h/w} \cdot 52\ \mathrm{m} = 1820\ \mathrm{h/a} \tag{2}$$

$$1820\ \mathrm{h/a} - [(30 + 11 + 15)\mathrm{d} \cdot 35\mathrm{h/d}] = 1498\ \mathrm{h/a} \tag{3}$$

Werden die Jahrespersonalkosten von 54.720,00 € durch die Jahresarbeitszeit von 1498 Stunden dividiert, ergibt dies Stundenkosten für den Arbeitgeber in Höhe von 36,53 €. Da es sich hierbei um eine Schätzung handelt, wird vereinfacht von 40,00 € pro Stunde ausgegangen.

$$54.720{,}00\,€/a \div 220\ \mathrm{h/a} = 36{,}53\,€/h \approx 40{,}00\,€/h \tag{4}$$

Für Auszubildende mit einer Ausbildungsvergütung in Höhe von 1.207,00 € [5] ergeben sich nach selber Kalkulation Personalkosten in Höhe von 10,57 €, vereinfacht 10,00 € pro Stunde.

[3] Vgl. IGMETALL-BAYERN.DE
[4] Vgl. Dr. Peter Hoberg
[5] Vgl. IGMETALL.DE

A.2 Installationsskript für Ubuntu 18.04

```
# Systemupdate
apt update
apt dist-upgrade


# Grundinstallation icinga und MySQL
apt install icinga2 icingacli monitoring-plugins mysql-server mysql-client
mysql_secure_installation


# Installation Icinga IDO und Web-Interface
apt install icinga2-ido-mysql
apt install icingaweb2


# Klonen Github-Repository
cd /usr/share
mv icingaweb2 old.icingaweb2
git config –global http.proxy http://webproxy1.kuka.int.kuka.com:80
git clone https://github.com/Icinga/icingaweb2.git


# Installation fehlendes PHP-Plugin
apt install php7.2-curl
systemctl restart apache2


# Hinzufügen MySQL User für icingaweb2
mysql -e "CREATE USER 'icingaweb'@'localhost' IDENTIFIED BY '• • •';"
mysql -e "GRANT ALL PRIVILEGES ON *.* TO 'icingaweb'@'localhost';"
mysql -e "FLUSH PRIVILEGES;"


# Einrichtungstoken erstellen
icinga2 feature enable command ido-mysql
icingacli setup token create
systemctl restart icinga2
```

A.3 Screenshots

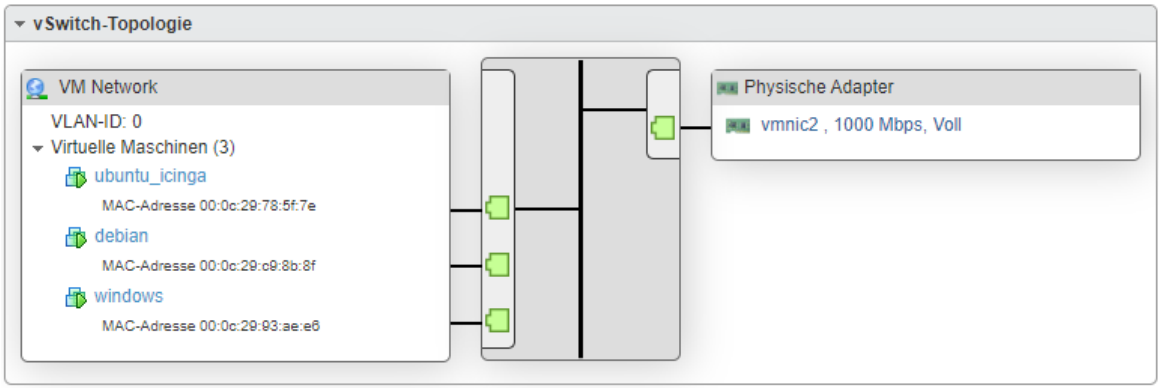

Abbildung 1: Netzwerkkonfiguration in VMware ESXi. (Rechts physische Netzwerkport, links virtuelle Ports der VMs, mitte ein virtueller Switch)

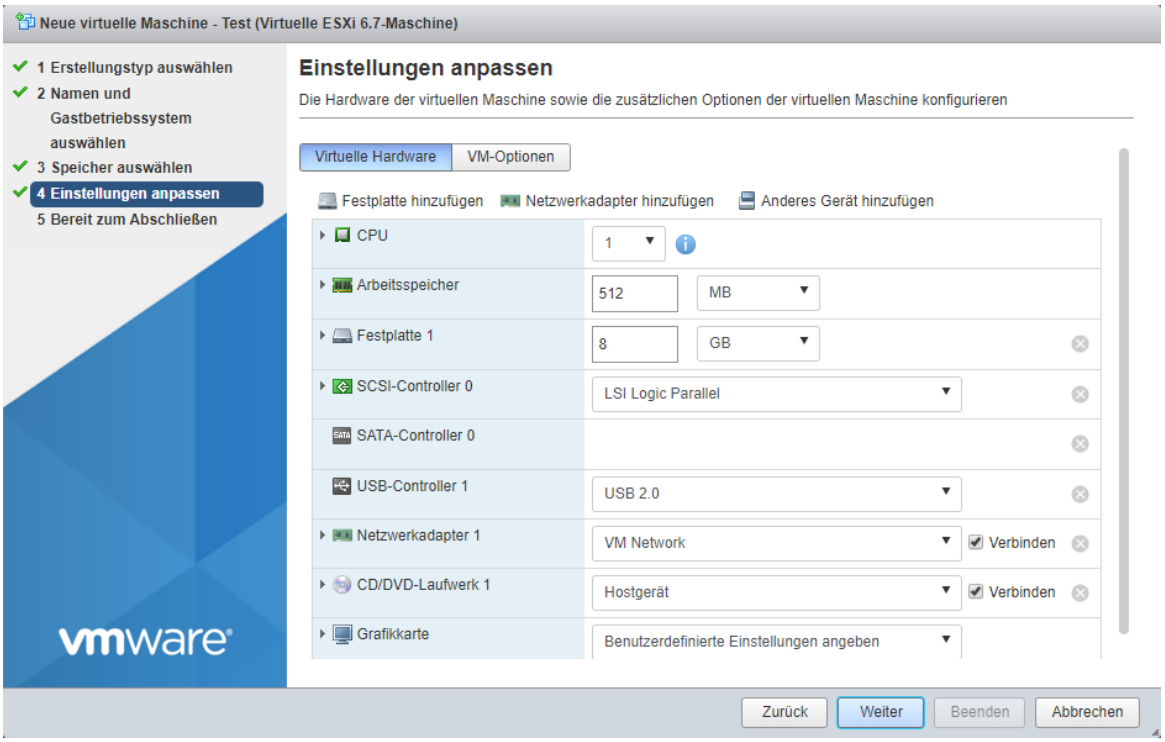

Abbildung 2: Parameteranpassung bei Erstellung einer virtuellen Maschine in VMware ESXi

```
administrator@ubuntu_icinga:/etc/icinga2/conf.d$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD PLUGIN can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD plugin?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW    Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary                  file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 2
Please set the password for root here.

New password:

Re-enter new password:

Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : y
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.


Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.


Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y
 - Dropping test database...
Success.

 - Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
```

Abbildung 3: Das Skript „mysql_secure_installation“ zur Absicherung eines MySQL-Systems

Abbildung 4: PHP-Fehler nach Installation des Icinga-Webfrontends

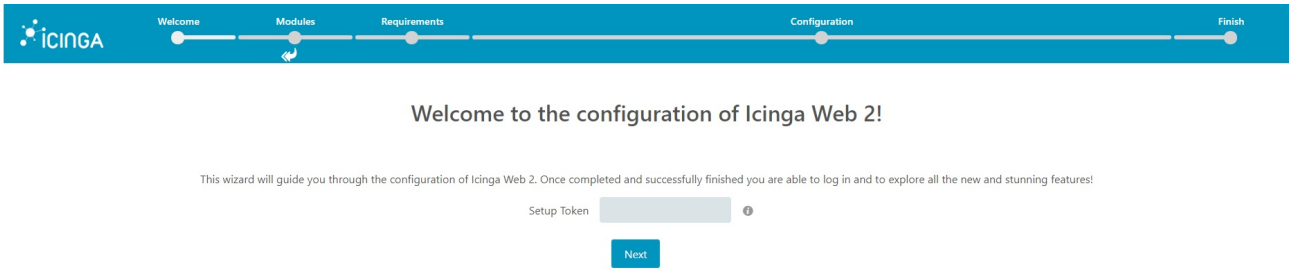

Abbildung 5: Willkommens-Bildschirm des Konfigurationsassistenten

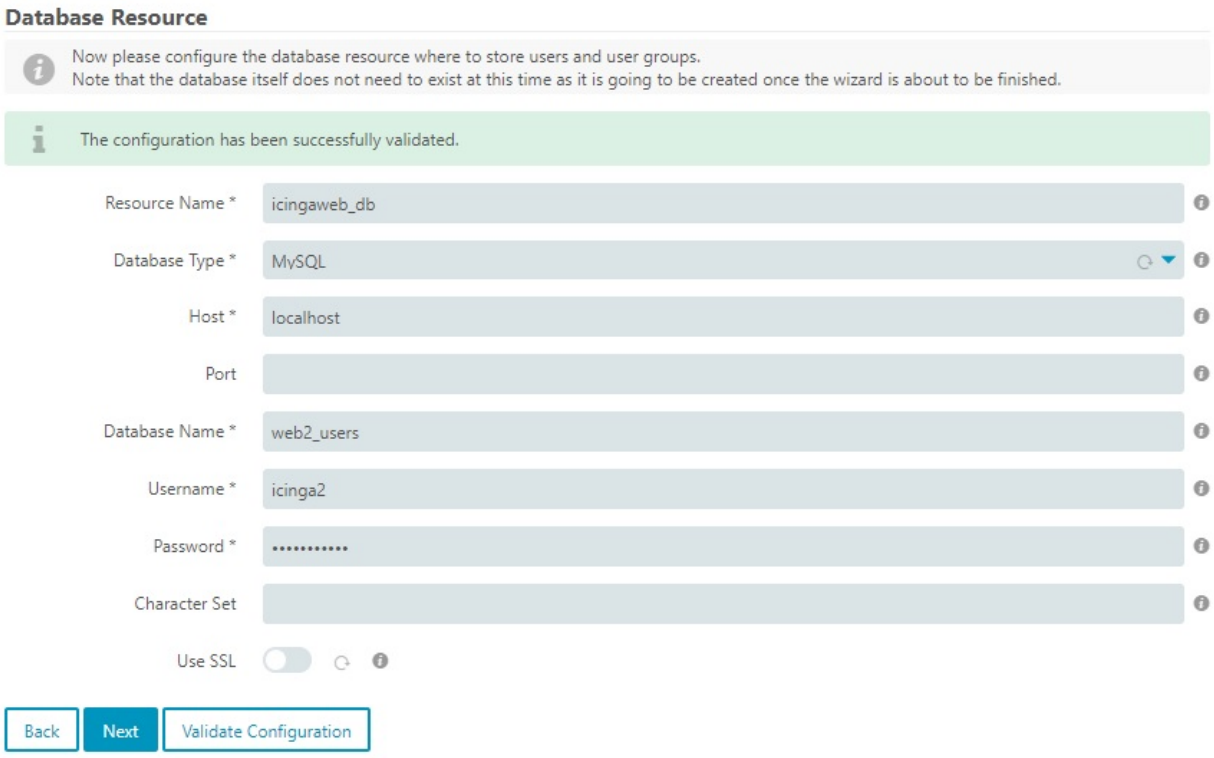

Abbildung 6: Einrichtung der Datenbank für Webfrontend-Benutzer

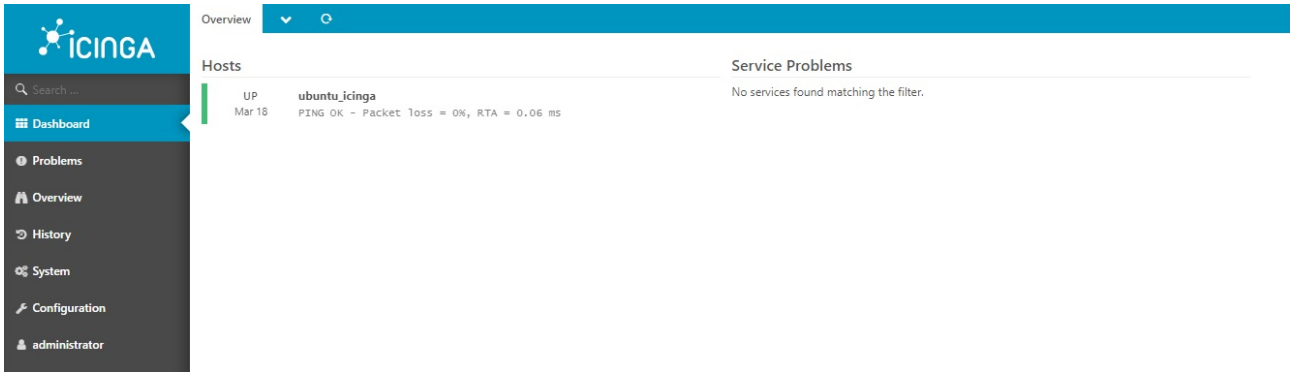

Abbildung 7: Startseite von „Icinga 2“ nach der Erstkonfiguration. Der Server, auf dem die Instanz von „Icinga 2“ läuft, ist automatisch als erster Server hinzugefügt

```
object Host "debian" {                  // Name des Hosts, am besten FQDN
  import "generic-host"                 // Vorlage für Host
  address = "10.129.37.217"             // IP-Adresse oder FQDN
  vars.os = "Linux"                     // Betriebsystem-Variable
  vars.agent_endpoint = "debian"        // Endpunkt auf dem die Checks durchgeführt werden
                                        // In diesem Fall das hiermit erstellte Host object "debian"
}

object Host "windows" {
  import "generic-host"
  address = "10.129.37.76"
  vars.os = "Windows"
  vars.agent_endpoint = "windows"
}
```

Abbildung 8: Konfiguration in `/etc/icinga2/conf.d/hosts.conf` um zwei neue Server dem Monitoring hinzuzufügen

```
object CheckCommand "cpuload" {
 command = [ PluginDir + "/check_load" ]
 arguments = {
  "-r" = { }

  "--critical" = {
   value = "0.8,0.8,0.8"
  }
 }
}
```

Abbildung 9: Beispiel für ein definiertes „CheckCommand“-Objekt

```
apply Service "loadcheck" {
 import "generic-service"

 check_command = "cpuload"

 command_endpoint = host.vars.client_endpoint

 assign where host.address
}
```

Abbildung 10: Beispiel für einen definierten Dienst

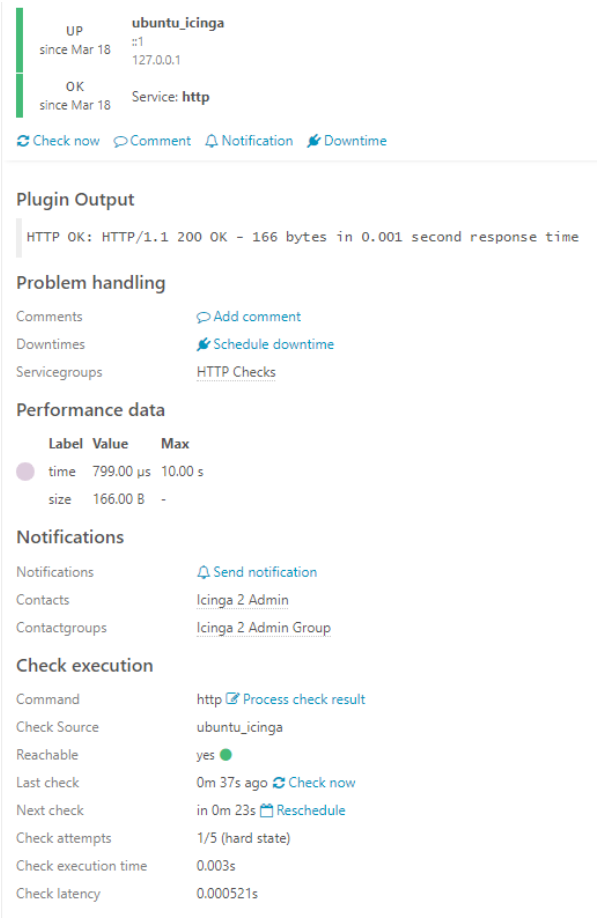

Abbildung 11: Detailansicht eines überprüften Dienstes; hier die Webserverüberwachung

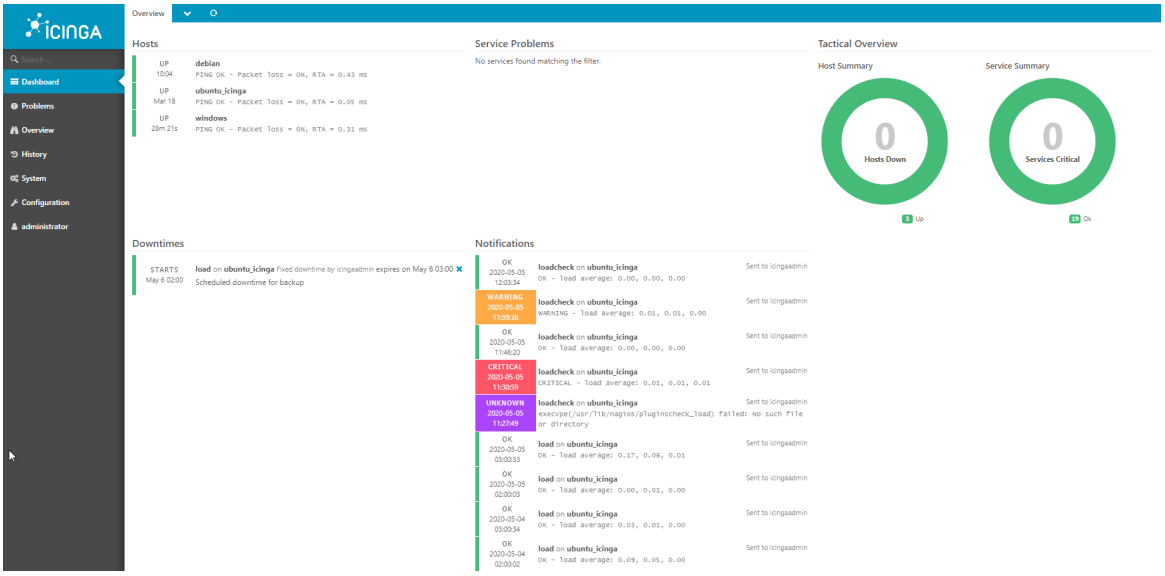

Abbildung 12: Selbsterstelltes Dashboard

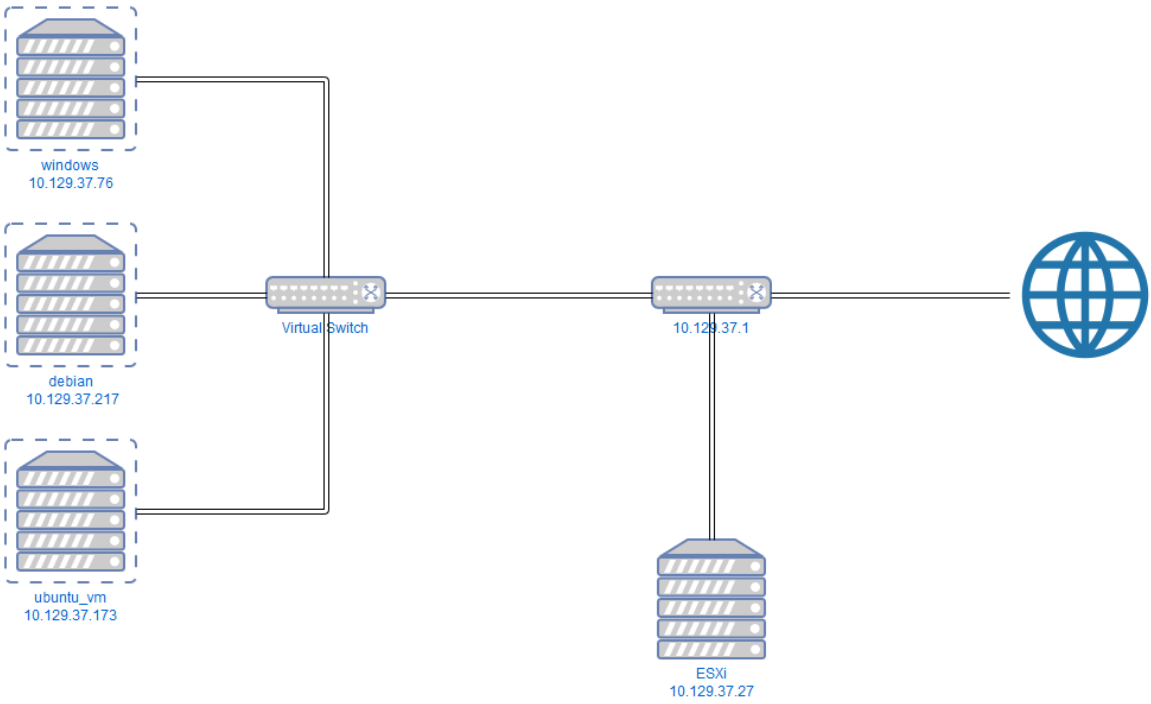

Abbildung 13: Netzwerkplan des Projektaufbaus