

Entwicklung und Evaluierung eines umfassenden Monitoring-Konzepts für die IT-Infrastruktur eines Lehrstuhls unter Verwendung aktueller Open-Source-Lösungen.

Problemstellung:

Die IT-Infrastruktur von Universitätslehrstühlen wird zunehmend komplexer, umfasst eine Vielzahl von Systemen (Server, Netzwerke, Storage, Workstations, Applikationen und Dienste) und ist essenziell für Forschung, Lehre und Verwaltung. Ein Ausfall oder eine Performance-Einschränkung kritischer Dienste kann den Lehrbetrieb massiv stören. Aktuelle Monitoring-Lösungen sind oft proprietär, kostenintensiv oder nicht optimal auf die spezifischen heterogenen Anforderungen und knappen Ressourcen eines akademischen Umfelds zugeschnitten.

Ziel dieser Arbeit ist es, die spezifischen Anforderungen an ein modernes, robustes und ressourcenschonendes IT-Monitoring für den Lehrstuhl zu ermitteln. Darauf aufbauend sollen geeignete Open-Source-Systeme (z.B. basierend auf der ELK-Stack, Prometheus/Grafana oder Nagios-Derivaten) identifiziert, bewertet und in einem Pilotaufbau implementiert werden. Dies soll die Proaktivität bei der Fehlerbehebung verbessern, Performance-Engpässe frühzeitig erkennen und langfristig die Betriebsstabilität der Lehrstuhl-Infrastruktur sicherstellen.

Aufgabenspektrum:

- Analyse und Konzeption (Task 1): - Literaturrecherche zu aktuellen Best Practices und Architekturen im Bereich IT-Infrastruktur-Monitoring (Netzwerk, Server, Applikationen, Logs, Security).
 - Erhebung spezifischer Anforderungen an das Monitoring am Lehrstuhl (Welche Systeme sind kritisch? Welche Metriken müssen erfasst werden? Welche Alarmierung ist notwendig?).
 - Entwicklung eines Konzepts für die Architektur des Monitoringsystems (Datenquellen, Datensammlung, Speicherung, Visualisierung, Alarmierung).
 - Systemauswahl und -bewertung (Task 2):
 - Identifikation und Vergleich potenzieller Open-Source-Monitoring-Lösungen (z.B. Icinga, Prometheus/Grafana, Zabbix, ELK-Stack, TICK-Stack).
 - Erstellung eines Bewertungskatalogs anhand der zuvor erarbeiteten Anforderungen (Skalierbarkeit, Wartbarkeit, Ressourcenverbrauch, Funktionsumfang).
 - Implementierung und Evaluierung (Task 3):
 - Aufbau eines Proof-of-Concept (PoC) anhand der präferierten Lösung, exemplarisch integriert in die Lehrstuhl-Infrastruktur (z.B. Monitoring von einem Server, einem Netzwerkgerät und einem kritischen Dienst/Logfile).
 - Konfiguration und Anbindung verschiedener Datenquellen (SNMP, System-Agenten, Log-Files).
 - Erstellung von Dashboards zur Visualisierung und Konfiguration von Alerts für kritische Zustände.
 - Evaluierung des implementierten Systems anhand der Anforderungen und kritische Dokumentation der Ergebnisse und Empfehlungen für den produktiven Einsatz.

Anforderungen:

- Fundierte Kenntnisse im Bereich der allgemeinen IT-Infrastruktur (Server-Betriebssysteme - Linux/Windows, Netzwerkprotokolle, Virtualisierung).
 - Programmierkenntnisse in mindestens einer Skriptsprache (z.B. Python oder Bash) sind wünschenswert.
 - Erfahrung oder Bereitschaft, sich intensiv in Monitoring-Tools (z.B. Prometheus, Grafana, Zabbix, ELK) und deren Konfiguration einzuarbeiten.
 - Selbstständige, strukturierte und ergebnisorientierte Arbeitsweise.
 - Gute Deutsch- und/oder Englischkenntnisse.

Vorab Literatur: - Site Reliability Engineering: How Google Runs Production Systems (für die konzeptionelle Grundlage: SLIs/SLOs). - Practical Monitoring: Effective Strategies for the Real World (Mike Julian)

Schwierigkeitsgrad: Master

Arbeit/Betreuung auf Deutsch oder Englisch möglich.

Kontakt: Dennis.Hartmann@informatik.uni-augsburg.de