

FISI-Antrag für das betriebliche Abschlussprojekt

Von Lisa Dibowski und Simon Mentel

Projektbezeichnung:

Automatisieren von Amazon Web Service Instanzen.

1.1 Kurzform der Aufgabenerstellung:

Für die Placeholder GmbH soll der Workflow beim Aufsetzen und Bereitstellen von Amazon Web Service Instanzen stark optimiert werden.

Hierzu soll eine größere Anzahl von Skripten konzipiert und entwickelt werden, mit denen das interne Team Kundenwünsche effizient und zielgerichtet umsetzen kann.

Dadurch wird das Skalieren und Verwalten zukünftiger und bestehender Instanzen merkbar vereinfacht.

1.2 Ist-Analyse

Aufgrund wachsender Ansprüche von Kunden, die das Unternehmen Placeholder GmbH in letzter Zeit gewonnen hat, wurde sich vor sechs Monaten entschieden, vermehrt in den Cloud Bereich zu gehen, um verschiedene Services bereitzustellen. Interne Gespräche haben dazu geführt, dass ein neues Team gebildet wurde, dass aktuell dafür verantwortlich ist, Instanzen in der Amazon Web Service Management Konsole zu erstellen und von dieser aus zu konfigurieren. Ein typischer Ablauf ist also, dass neue Aufträge und Änderungswünsche an Firmeninterne Systeme gesendet werden und dort in einem Ticket erfasst und verteilt werden. Das führt dazu, dass einzelne Mitarbeiter teilweise lange Zeit mit simplen Aufgaben beschäftigt sind, die eigentlich zentral über Skripte automatisiert werden könnten. Auch der Sicherheitsaspekt ist momentan nicht optimal, da sich mehrere Mitarbeiter Security Schlüssel teilen und diese lokal auf ihren Arbeitsmaschinen liegen. Dieses Vorgehen führt dazu, dass von Zeit zu Zeit neue Root Schlüssel generiert und an mehrere Mitarbeiter verteilt werden müssen, was ebenfalls einschränkt in der Effizienz mit sich führt. Durch die fehlende Automatisierung ist es des Weiteren schwierig, einen allgemeinen Standard zu gewährleisten. Vereinzelte Mitarbeiter brauchen verschieden viel Zeit, um Aufgaben zu bewältigen, ein gemeinsam geführtes Repository könnte hierbei sehr von Nutzen sein. Diese und weitere Zustände haben dazu geführt, dass es bei der steigenden Anzahl und Ansprüchen von Kunden zu langen Wartezeiten kommt, worunter die normalen hohen Standards der Firma leiden. Jede weitere Instanz und jeder weitere Kunde bringt also erhebende Schwierigkeiten und komplexe Problembewältigung mit sich.

2. Zielsetzung entwickeln / Soll-Konzept

Durch verschiedene Skripte soll es zukünftig ermöglicht werden, ohne großen Zeitaufwand Änderungen und Konfigurationen von AWS-Instanzen durchzuführen. Viele der Kunden der Placeholder GmbH benutzen AWS-Instanzen zum Speichern wichtiger Daten. Ein manueller file upload bei 5 verschiedenen S3 Buckets ist noch ohne weiteres möglich, wenn es sich bei größeren Kunden allerdings um mehrere hundert S3 Buckets handelt, sollen zukünftig interne Skripte für diese Aufgabe zum Einsatz kommen. Bei anderen Kunden werden AWS EC2 (Amazon Elastic Compute Cloud) Instanzen eingesetzt, um dynamische Rechenleistung zur bereitzustellen. Momentan wird das Aufsetzen und zur Verfügung stellen dieser Instanzen noch manuell durchgeführt. Weitere zu automatisierende Services sind Amazon CloudWatch, was von vielen Kunden zum Eventlogging und das Stellen von Alarmen benutzt wird. Im Webhosting Bereich sollen Skripte in Zukunft helfen, die Konfiguration des relevanten S3 Buckets abzurufen, beziehungsweise neue Konfigurationen zu setzen. Auch das Verschlüsseln von wichtigen sensiblen Daten soll in Zukunft automatisiert über den AWS Key Management Service geregelt werden.

2.2. Welche Anforderungen müssen erfüllt sein?

Folgende Anforderungen sollen durch die Boto3 Skripte erfüllt werden:

- Unternehmensweite Standardmethoden zur Verwaltung der Instanzen
- Sicherheit durch verbessertes Key Management
- Der Python3 Code soll getestet sein und ohne Syntaxfehler
- Auslesen von Konfigurationen einzelner AWS-Instanzen
- Aufspielen von Konfigurationen bestehender und neuer AWS-Instanzen
- Vereinfachtes Logging ohne Einsatz der Webkonsole
- Einfache Aufgaben sollen dokumentiert und komplett automatisiert sein
- Skripte sollen vereinzelt durch Cron-Jobs regelmäßig ausgeführt sein
- Versionskontrollen und Updates dürfen nicht mehr manuelle Aufgabe der Mitarbeiter sein
- Der Code muss gut dokumentiert und zentral für die Mitarbeiter zur Verfügung stehen

2.3 Welche Einschränkungen müssen, berücksichtigt werden?

Da sich die Placeholder GmbH sich auf AWS für Cloud-Computing-Services spezialisiert hat, werden die Automatisierungen für AWS geschrieben. Andere Anbieter für Cloud-Computing-Services werden in diesem Projekt nicht berücksichtigt. Die Automatisierungsskripte werden mithilfe von Python 3 geschrieben, sollte es in der Zukunft neue Technologien und neuen Programmiersprachen geben, müssen neue Automatisierungen geschrieben werden.

3. Projektstrukturplan entwickeln

3.1. Was ist zur Erfüllung der Zielsetzung erforderlich?

Das Projekt soll agil entwickelt werden. Hierzu wird in kurzen Iterationszyklen Rücksprache mit dem Auftraggeber gehalten, um Feedback einzuholen. Dieses Feedback ermöglicht den Entwicklern, schnell auf Änderungswünsche einzugehen und somit Zeit zu sparen. Durch die ständige Einbindung der Endanwender wird auch ermöglicht, dass eine Schulung zur Benutzung der Automatisierungen verkürzt wird, da alle Features und Änderungen schon bekannt sind. Das Fortschritt am Projekt wird mithilfe eines Scrum boards dokumentiert.

3.2 Aufgaben auflisten

Analyse:

- Durchführung einer Ist-Analyse
- Ermittlung von Use Cases von AWS-Services
- Erstellen eines Soll-Konzepts
- Beurteilung der Möglichkeiten und erforderlichen Werkzeuge
- Erstellung eines Lastenhefts

Entwurf:

- AWS Test Instanzen konzipieren
- Planung der Umsetzung durch studieren der AWS-Dokumentation
- Planung der Skripte durch studieren der Boto3 Dokumentation
- Erstellung eines Pflichtenhefts
- Planen von Lambda Funktionen

Implementierung:

- AWS Test Instanzen werden aufgesetzt
- Benötigte Skripte werden geschrieben und dokumentiert
- Wichtige Sicherheitsfeatures von Amazon werden umgesetzt
- Implementierung und testen von einfachen Session Skripts
- Implementierung und testen von komplexeren Skripts für verschiedene use cases
- Implementierung der Lambda Funktionen
- Aufsetzen benötigter Cron-Jobs

Dokumentation:

- Erstellung einer Projektdokumentation
- Erstellung der Entwicklerdokumentation
- Erstellung der Benutzerdokumentation

4. Projektphasen mit Zeitplanung in Stunden

Analyse	6 h
Durchführung einer Ist-Analyse	2 h
Ermittlung von Use Cases von AWS-Services	1 h
Erstellen eines Soll-Konzepts	1 h
Beurteilung der Möglichkeiten und erforderlichen Werkzeuge	1 h
Erstellung eines Lastenhefts	1 h
Entwurf	11 h
AWS Test Instanzen konzipieren	1 h
Planung der Umsetzung durch studieren der AWS-Dokumentation	2 h
Planung der Skripte durch studieren der Boto3 Dokumentation	1 h
Erstellung eines Pflichtenhefts	3 h
Planen von Lambda Funktionen	4 h
Implementierung	39 h
AWS Test Instanzen werden aufgesetzt	10 h
Benötigte Skripte werden geschrieben und dokumentiert	7 h
Wichtige Sicherheitsfeatures von Amazon werden umgesetzt	2 h
Implementierung und testen von einfachen Session Skripten	5 h
Implementierung und testen von komplexeren Skripten für verschiedene use cases	5 h
Implementierung der Lambda Funktionen	5 h
Aufsetzen benötigter Cron-Jobs	5 h
Dokumentation	9 h
Erstellung einer Projektdokumentation	7 h
Erstellung der Entwicklerdokumentation	1 h
Erstellung der Benutzerdokumentation	1 h

5. Name der Ausbildungsstätte in dem das Projekt durchgeführt wird

Berufliche Schule ITECH Elbinsel Wilhelmsburg (BS14)

Dratelnstraße 26

21109 Hamburg