



## KN02: IaaS - Virtuelle Server

- A) AWS Kurs (30%)
  - a) Lab 4.1 - EC2
  - b) Lab 4.2 - S3
- B) Zugriff mit SSH-Key (20%)
- C) Installation von Web- und Datenbankserver (50%)

# KN02: IaaS - Virtuelle Server

---

Beachten Sie die [allgemeinen Informationen zu den Abgaben](#).

Wir werden nun mit einem public cloud Anbieter arbeiten. Wir verwenden Amazon Web Services (AWS). Aber alles was wir hier umsetzen, bieten auch andere public cloud Anbieter (z. B. Microsoft Azure).

Wir verwenden teilweise die AWS Schulungsumgebung für diese Kompetenz. Sie haben eine Einladung für zwei Kurse bekommen, die wir folgend verwenden werden.

Ein paar zusätzliche notwendige Grundlagen finden Sie in der Datei [Cloud Computing in Gitlab](#).

Sie erlernen den grundsätzlichen Umgang mit AWS. Anschliessend werden Sie lernen wie sie sich mit einem SSH Key einloggt, gefolgt vom Verständnis für Cloud-init. Am Schluss vermischen Sie SSH-Key mit cloud-init Konfiguration.

## A) AWS Kurs (30%)

Öffnen Sie den Kurs **"AWS Academy Introduction to Cloud: Semester 1"** und arbeiten Sie sich durch die ersten 4 Module (Global Infrastructure, Structures of the Cloud, AWS Console, Virtual Servers). Sie müssen die Knowledge Checks **nicht** durchführen, dürfen aber.

Die ersten drei Module sind informativ und sollten **schnell durchgegangen werden**. Das vierte Modul sind praktische Übungen und werden mit den folgendem Teil a) und b) bearbeitet. Sie lernen grundsätzlich mit virtuellen Maschinen umzugehen.

### a) Lab 4.1 - EC2

Führen Sie alle Schritte durch, **aber nicht das Lab beenden**. Sie erstellen die Screenshots der verschiedenen Bereiche für Ihre Abgabe **nachdem** Sie die Schritte des Moduls durchgeführt haben.

**Warnung:** Das Lab schliesst nach 1.5 Stunden und ihr Fortschritt wird komplett gelöscht. Führen Sie diese Teilaufgabe a) in einem Zug durch.

Screenshots:

- HTML-Seite, inkl. URL

- Liste der EC2-Instanzen
- Details der *Web Server*-Instanz (öffentliche IP sichtbar)
- Security-Group: Liste der Inbound-Regeln.

## b) Lab 4.2 - S3

Führen Sie alle Schritte durch, **aber nicht das Lab beenden**. Sie erstellen die Screenshots der verschiedenen Bereiche für Ihre Abgabe **nachdem** Sie die Schritte des Moduls durchgeführt haben.

**Ausnahme:** Kopieren Sie Ihr JSON der Bucket-Policy in eine JSON-Datei für die Abgabe.

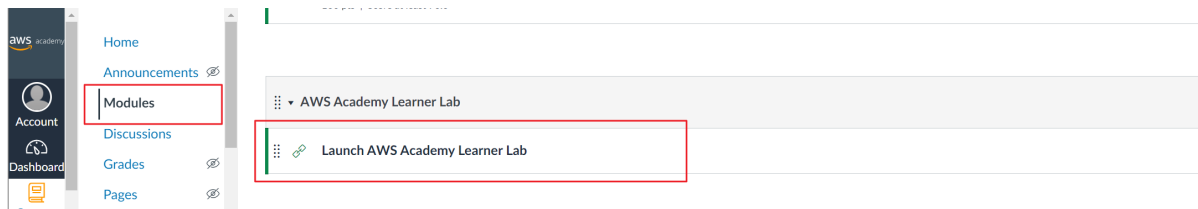
**Warnung:** Das Lab schliesst nach 1.5 Stunden und ihr Fortschritt wird komplett gelöscht. Führen Sie diese Teilaufgabe b) in einem Zug durch.

Screenshots:

- Liste der Buckets.
- HTML-Seite, inkl. URL.
- Liste der Dateien im Bucket.
- Eigenschaften von "Static website hosting".
- JSON-Datei mit der Bucket-Policy für den öffentlichen Zugriff

## B) Zugriff mit SSH-Key (20%)

Wechseln Sie nun den Kurs und verwenden "**AWS Academy Lerner Lab**". Ab sofort arbeiten Sie **immer** im **Lerner Lab**, ausser explizit anders angegeben. Im Lerner Lab werden Fortschritte **nicht** gelöscht, aber die Umgebung fährt nach 4h herunter und Sie müssen Sie neu starten.



**Wichtig:** [Lesen Sie sich in die funktionsweise von SSH und Private/Public Key ein.](#)

Erstellen Sie nun wieder eine neue **Ubuntu**-Instanz in AWS mit den **Standard**-Einstellungen und den folgenden **zusätzlich**:

- Bei "Key pair" erstellen Sie **zwei** Key-Value Pairs! Geben Sie Namen wie <Ihr-Name>-1 und <Ihr-Name>-2. Stellen Sie sicher, dass Sie beide Schlüssel herunterladen.
- Bei "Key pair" wählen Sie dann den **ersten** Key aus um weiterzufahren.
- Starten Sie die Instanz!

Zeigen Sie, dass sie sich **nur** mit **dem ausgewählten** Schlüssel einloggen können.

**Hinweis:** AWS speichert ihren privaten Schlüssel **nicht**. Sie können den nicht wiederherstellen. Auf dem Ubuntu-Server liegt immer nur der öffentliche Schlüssel. Seien Sie also vorsichtig und verlieren Sie den Schlüssel nicht.

**Abgaben:**

- Screenshot mit dem ssh-Befehl und des Resultats unter Verwendung des **ersten** Schlüssels

- Screenshot mit dem ssh-Befehl und des Resultats unter Verwendung des **zweiten** Schlüssels
- Screenshot der Instanz-Detail (oder Liste), so dass der verwendete Schlüssel sichtbar ist.

## C) Installation von Web- und Datenbankserver (50%)

Verwenden Sie die Instanz aus Aufgabe B). Wir installieren nun einen Webserver und eine Datenbank auf ihrem Ubuntu. Führen Sie folgende Punkte aus:

- Offene Ports:
  - 22 für SSH Zugriff
  - 80 für HTTP Zugriff
- Führen Sie folgende Befehl aus. Sie müssen die Befehle **nicht** dokumentieren.
  - `sudo apt update` auf. Dieser Befehl aktualisiert die Pakete/Software.
  - `sudo apt install apache2`. Dieser Befehl installiert den Apache Webserver.
  - `sudo apt install php`. Dieser Befehl installiert den PHP Applikationsserver.
  - `sudo apt install libapache2-mod-php`. Dieser Befehl installiert die PHP-Erweiterung für Apache.
  - `sudo apt install mariadb-server`. Dieser Befehl installiert den Datenbankserver.
  - `sudo apt install php-mysqli`. Dieser Befehl installiert das PHP-Modul, um Datenbank-Abfragen zu erstellen.
  - `sudo mysql -sfu root -e "GRANT ALL ON *.* TO 'admin'@'%' IDENTIFIED BY 'password' WITH GRANT OPTION;"`. Dieser Befehl erstellt einen neuen Benutzer *admin* mit dem Passwort *password*. Wir werden diesen gleich verwenden.
  - `sudo systemctl restart mariadb.service`. Dieser Befehl startet den DB Server neu.
  - `sudo systemctl restart apache2`. Dieser Befehl startet den Webserver neu.
- `cd ~`. Dieser Befehl stellt sicher, dass Sie sich in ihrem Home-Verzeichnis befinden
- `git clone https://gitlab.com/ch-tbz-it/stud/m346/m346scripts.git`. Diesen Befehl sollten Sie bereits beherrschen.
- `sudo cp ./m346scripts/KN01/*.php /var/www/html/`. Dieser Befehl kopiert die beiden PHP-Dateien in das Verzeichnis des Webserver.
- Rufen Sie von ihrem Rechner nun die folgenden Seiten auf:
  - `http://[Ihre-IP]/index.html`. Diese URL zeigt ihnen die Informationen von Apache an. Wenn die Seite funktioniert, zeigt dies, dass Apache installiert wurde und funktioniert.
  - `http://[Ihre-IP]/info.php`. Diese URL zeigt ihnen die Informationen von PHP an. Wenn diese Seite funktioniert, zeigt dies, dass das PHP-Modul funktioniert und Sie nun dynamische Seiten hosten können.
  - `http://[Ihre-IP]/db.php`. Diese URL zeigt ihnen die Datenbank-Benutzer. Wenn diese Seite funktioniert, zeigt es Ihnen, dass die Datenbank-Verbindung funktioniert.

**Hinweis:** Falls sudo nicht geht, können Sie `yum` verwenden. Vollständige Erklärung zu `yum` siehe: <https://elsefix.com/de/el-comando-yum-en-linux-una-referencia-completa.html>. Allerdings haben Sie in diesem Fall das falsche Image ausgewählt (anstatt Ubuntu amazon-linux). Entweder Sie korrigieren diesen Fehler oder Sie verwenden alternativ die folgende Anleitung: <https://linux.how2shout.com/installing-mariadb-on-amazon-linux-2023/>

**Abgabe:**

- Erstellen Sie Screenshots der **funktionierenden** Webseiten (mit sichtbarer URL)