Programmieren einer Webseite auf der man C++-Files schreiben und ausführen kann

Nicola Krull

Inhaltsverzeichnis

1 Abstract

2 Einleitung

3 Aufbau

3.1 Design

3.2 Software

Das Ziel des Projektes ist eine dynamische Webseite zu bauen. Eine dynamische Webseite ist eine Webseite bei welcher der Server mit der Webseite kommuniziert.[**01**] Für den Bau einer dynamischen Webseite benötigt man ein Webframework, welches die Interaktionen zwischen den Files steueren kann. Dafür wird das Webframework *Express* verwendet. Es ist ein serverseitiges Webframework, welches für die Plattform *Node.js* entwickelt wurde.[**02**] *Node.js* ist eine open-source, serverseitige Plattform, welche *JavaScript* als Skriptsprache verwendet.[**03**] Wenn man eine dynamische Webseite programmiert muss man immer zwischen serverseitigen und clientseitigen Code unterscheiden. Für die serverseitigen Programme verwendet man, die Plattform *Node.js* und auf der Clientenseite läuft *JavaScript*, *HTML* und *CSS*-Code. Dabei wird in diesem Fall die Templatesprache *Jade* verwendet, welche zur Generierung von *HTML*-Seiten zuständig ist. Es vereinfacht nicht nur die Syntax, sondern es funktion wie anderen Programmiersprachen, somit kann man für den *HTML*-Code Variabeln, If-Abfragen und for-Schleifen benützen. [**04**]

Express kann man in die drei Teile, views, routes und controllers aufteilen. In dem views-Ordner sind die Jade Files abgeschrieben. Diese File sind für das Design der Webseite zuständig. Also wenn man wie zum Beispiel eine Textbox und einen Button auf der Webseite sehen möchte, müsste man in einem *Jade*-file eine Textarea und einen Button kreieren. Durch das coden dieser Elemente hat man eine Textbox und einen Button auf der Webseite aber es würde keine reaktion geben wenn man in die Textboxt hinein schriebt oder den Button drückt. Man möchte den Textinhalt weiterleiten. Dafür benötigt man die HTTP Methoden wie zum Beispiel GET und POST. Die Abkürzung HTTP steht für Hypertext Transfer Protokol und ist zuständig für die Kommunikation zwischen dem Client und dem Server.[**06**] All diese HTTP Request werden im routes Ordner bearbeitet und zum Controllers-file weitergeleitet. In den Controllers-Files findet alle Actionen statt. Alles was vom Server berechnet oder ausgeführt werden muss. In meinem Fall wäre das das Compilieren von den Textdaten in der Textbox. Dafür hat man Child Processes von *Node.js* benutzt. Mit Child Processes können die Terminalbefehle automatisch ausgeführt werden und somit muss man es nicht manuel eintippen. Da man keine Datenbanken für meine Arbeit verwende, ist der gebrauch von Models nicht nötig.

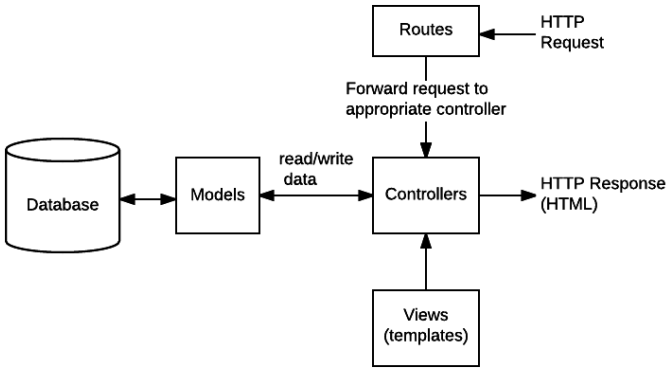

Abbildung 1: Express Aufbau [**05**]

4 Security

4.1 Sicherheitsprobleme

4.2 Mögliche Sicherheitsansätze

4.2.1 geplannter Ansatz

Eine andere Lösung wäre ein System in welchen die Dateiberechtigungen verändert wird und die eingehenden und ausgehenden Ports schliessen würde. In Unix Systeme kann man mit dem Befehl chmod die Berechtigungen von Ordnern und Dateien verändern. Dabei gibt es drei Gruppen Eigentümer(user), Gruppe(group) und Sonstige(others). Für jede dieser Gruppen kann man drei verschiedene grund Rechte bestimmen. Das erste Recht **r** steht für den englischen Begriff read und bewirkt dass man das Recht hat das File zu lesen. Das zweite Recht wird mit dem Buchstaben **w** ausgedrückt. Es steht für write und ermöglicht dem Benutzer Dateien zu schreiben. Das heisst er darf Dateien und Unterverzeichnisse erstellen, löschen, ändern und die Dateirechte der Unterverzeichnisse verändern. Der letzte Buchstabe ist **x** und steht für execute. Diese Funnktion führt Dateien als Programme aus. Im Falle eines Verzeichnis hätte man die Berechtigung in das Verzeichnis hineinzugehen und so die Dateien und Unterverzeichnisse zu erreichen. Meisst benutzt man eine kombination der drei Berechtigungsfunktionnen. Die einzelnen Berechtigungsfunktionen sind zu einer Zahl angeordnet. Das **x** ist gleich eins, das **w** hat den Wert zwei und das **r** den Wert vier. Wenn man eine kombination der drei Berechtigungen möchte muss man sie addieren. Im Falle, dass der Benutzer alles machen möchte, also lesen, schreiben und ausführen muss man die Berechtigung auf die Nummer sieben legen, da $1(\mathbf{x}) + 2(\mathbf{w}) + 4(\mathbf{r}) = 7$ ist.

4.2.2 chroot

Chroot steht für "change rootünd es ist ein Programm welches das Rootverzeichnis für Unixsysteme ändern kann.[**07**] Unix ist ein Betriebssystem, welches in den sechziger Jahren entwickelt wurde. Viele Betriebsysteme basieren auf diesem System, unteranderem das macOS, das iOS und die Linux Betriebsysteme, dazu gehört auch das Betriebsystem Android.[**08**] Es generiert eine geschlossene Umgebung namens chroot jail. Diese Umgebung erlaubt den Zugriff auf Files und Befehle ausserhalb dieses Ordners nicht. Somit kann der Users nur auf diesem bestimmten Breich des Servers zugreiffen und somit keinen Schaden an dem Server anrichten.[**09**]

4.2.3 Containervirtualisierung

Containervirtualisierung ist ein Verfahren, welches sich auf eine Betriebssystemsfunktion bezieht, bei der der Kernel die Erstellung von multiplen isolierende User-Space Instanzen erlaubt. Diese erstellten Instanzen werden Containers gennant.[**10**] Ein Kernel ist die tiefste Softwareschicht eines Betriebsystem. Der Kernel ist zuständig für die Prozess- und Datenorganisation. Ausserdem kann der Kernel direkt auf die Hardware zugreifen.[**11**] Die virtuelle Speicherverwaltung wird in User-Space und Kernel-Space unterteilt. Der User-Space ist der Ort an dem die Anwendungssoftwaren ausgeführt werden. [1] Der Unterschied zwischen einem Programm welches in einem Container und eines das von einem Betriebssystem ausgeführt wurde ist, dass das Programm vom OS aus alle Elmente des Betriebssystem zur Verfügung hat, während das File im Container nur auf die Informationen innerhalb des Containers zugreifen kann. Da mann komplett isoliert ist muss man einige Elemnte wie zum Beispiel Libaries für die Programme, die in diesem Container ausgeführt werden, hinzufügen. Für unixartige Systeme sind Containers fortgeschrittene Implementierungen von chroot.[2]

Eine Containervirtualisierungstechnologie ist Linux Containers(Abkürzung LXC). Es ist ein Verfahren, welches eine Virtualisierung von Softwaren auf Betriebssystemebene innerhalb des Linux-Kernels generiert. Das Besondere an den Linux Containers ist, dass sie im Vergleich zu herkömmlichen Virtuelle Maschinen, wie zum Beispiel VMWare oder KVM, einzelne Anwendung in virtuellen Umgebugen ausführen können. Ausserdem ist es Möglich ein ganzes Betriebssystem in einem solchen Container zu starten. [3]

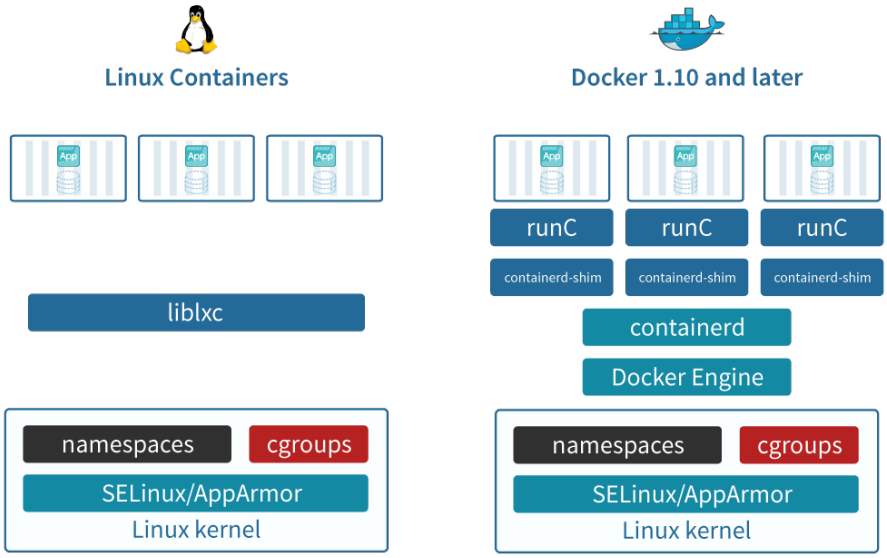

Abbildung 2: Architektur von zwei Containertechnologien [**15**]

Eine weitere Containervirtualisierungstechnologie ist Docker. Docker ist eines der bekanntesten Containervirtualisierungsarten. Früher basierte es auf dem LXC System. Man stellte dies jedoch ein und ersetzte den Teil der auf dem LXC System basierte durch eine eigene Implementierung names libcontainer. Genauergenommen ist es die Bibliothek liblxc (siehe Kapitel 4.2.5) die durch den libcontainer ersetzt wurde. Die liblxc Bibliothek wird immernoch von den Linux Containern eingesetzt. Im Linux Kernel sind praktisch alle Containertechnologien gleich aufgebaut. Dazu gehören die Kernel-Funktionen cgroups und Namespaces. Cgroups, auch Control Groups genannt, ist ein Verfahren welches die Ressourcen verwaltet. Diese Systemsoftware führt dazu das der Container nur eine begrenzte Menge an Speicher, Rechenleistung und Disk I/O zur Verfügung hat. Sonst würde er die ganzen Ressourcen für sich alleine beansprechen. Unter Disk I/O versteht man die Eingabe(Input) und Ausgabe(Output) Vorgänge einer physischen Festplatte. Wenn man eine Datei von der Festplatte lesen möchte muss der Prozessor warten bis die Datei gelesen wurde. Das selbe gilt auch fürs Schreiben. [**20**] Die Systemsoftware Namespace benötigt man um die Prozesse eines Containers zu verbegen. Ansonsten könnte man über einen weiteren kreierten Container Informationen über die Prozesse eines anderen Containers erhalten.[**21**] SELinux und

[1] https://en.wikipedia.org/wiki/User_*space* 31.12.2018

[2] https://en.wikipedia.org/wiki/Operating-system-level_*virtualization* 30.12.2018

[3] https://www.webhod.de/lxc-und-lxd-was-sind-linux-container 30.12.2018

APPArmor sind Mandatory Acces Control(MAC) Systeme. MAC Systeme sind zuständig für Zugriffe von Benutzer und Prozessen auf einzelnen Objekte. Wenn ein Zugriff nicht erlaubt ist wird er vom MAC System unterbunden.[**22**]

4.2.4 Hypervisor

Bei Hypervisoren, auch Virtual Machine Monitor gennant, handelt es sich um eine Software, welche virtuelle Maschinen von eiem beliebigen Betriebssystemen aus laufen lassen kann. Es gibt zwei Arten von Hypervisoren. [4]

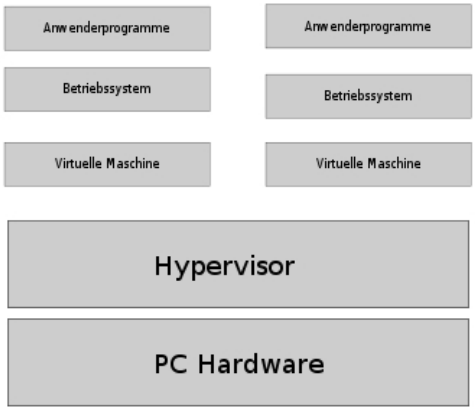

Abbildung 3: Typ-1-Hypervisor

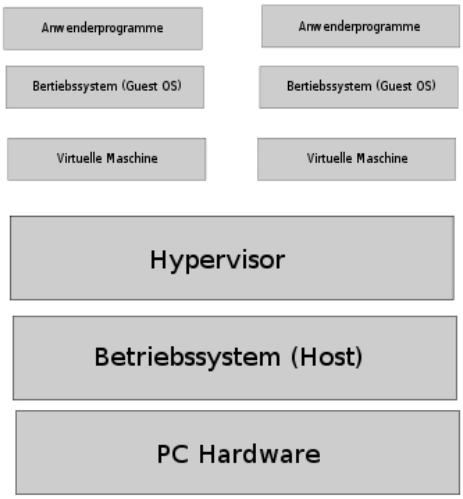

Abbildung 4: Typ-2-Hypervisor

Der Typ-1-Hypervisor auch native oder bare-metal Hypervisor genannt läuft direkt auf der Hardware und benötigt passende Treiber. Dieser Hypervisor benötigt kein vorherige Betriebssystem-Installation. Der Typ-2-Hypervisorbraucht ein vollständiges Betriebssystem auf dem Hostsystem. Es nutzt die Gerätetreiber des Benutzersystems für den Zugriff auf die Hardware des Hostsystems.

Das benutzen eines Hypervisor ist ein mögliche Sicherheitsansatz, da er virtuelle Umgebungen kreieren kann. Die Vorteile von diesen virtuellen Umgebungen sind, dass sie vollstäding isoliert sind und man kann immer wieder ein weiteres Betriebsystem kreieren oder löschen. Somit würde man ein Shell-Script schreiben, welches automatisch pro User eine virtuelle Umgebung kreiert. In dieser Umgebung würde dann auch die Compilierung stattfinden. Da es in einem virtuellen Betriebsystem stattfinden würde, würde nichts kaputt gehen. Nach dem Kompilieren, würde das Betriebsystem den Output in einen Ordner legen welcher das Hostsystem mit dem virtuellen Betriebsystem teilt. Sobald Output übermittelt wurde, wird das virtuele Betriebssystem gelöschen.

Ein möglicher Hypervisor ist der KVM. Es steht für Kernel-based Virtual Machine. Der KVM ist seit 2007 in den Linux Betriebtssysteme im Haupt-Kernel integriert.Damit es die Virtualisierung unterstützt muss es auf der x86 Hardware installiert sein.

4.2.5 liblxc

[4] https://www.searchdatacenter.de/definition/Hypervisor-Virtual-Machine-Monitor-VMM 10.1.2019

4.3 Angewendeter Ansatz

5 Textfunktionen

5.1 Syntax-Highlighting

5.2 andere

6 Erweiterungsmöglichkeiten

Abbildungsverzeichnis

Literatur

[1] https://blog.kompaktdesign.com/webdesign/statisch-vs-dynamisch 30.12.2018

[2] https://de.wikipedia.org/wiki/Express.js 25.10.2018

[3] https://de.wikipedia.org/wiki/Node.js 25.10.2018

[4] https://t3n.de/news/jade-638027/ 25.10.2018

[5] https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express$_{N}odejs/routes$ 19.01.2019

[6] https://www.w3schools.com/tags/ref$_{h}ttpmethods.asp$25.10.2018

[7] https://en.wikipedia.org/wiki/Chroot 30.12.2018

[8] https://de.wikipedia.org/wiki/Unix 30.12.2018

[9] https://wiki.archlinux.org/index.php/Chroot 30.12.2018

[10] https://en.wikipedia.org/wiki/Operating-system-level$_{v}irtualization$ 30.12.2018

[11] https://de.wikipedia.org/wiki/Kernel$_{(}Betriebssystem)$ 30.12.2018

[12] https://robin.io/blog/containers-deep-dive-lxc-vs-docker-comparison/ 23.01.2019

[13] http://blog.scoutapp.com/articles/2011/02/10/understanding-disk-i-o-when-should-you-be-worried 23.01.2019

[14] https://jaxenter.de/docker-einfuehrung-linux-basics-62049/6 23.01.2019

[15] https://www.admin-magazin.de/Das-Heft/2009/03/Mandatory-Access-Control-mit-Smack 23.01.2019