Serverdokumentation
Projekt: DigitalSchoolNotes

Projektgruppe: Adler, Brinnich, Hohenwarter, Karic, Stedronsky

Version 1.2 **12.10.2015** **Status: [RELEASE]**

	Datum	**Name**	**Unterschrift**
Erstellt	09.09.2015	Niklas Hohenwarter	
Geprüft			
Freigegeben			

Git-Pfad: /doc/serverdokumentation **Dokument:** serverdoc.tex

Inhaltsverzeichnis

1 Changelog

Version	Datum	Status	Bearbeiter	Kommentar
0.1	09.09.2015	Erstellt	Hohenwarter	Draft
0.2	21.09.2015	Bearbeitet	Hohenwarter	Alles verbessert
1.0	28.09.2015	QA	Stedronsky	Rechtschreibung
1.1	07.10.2015	Bearbeitet	Hohenwarter	Django Ports
1.1	12.10.2015	Bearbeitet	Hohenwarter	Deployment
1.2	14.10.2015	QA	Stedronsky	Rechtschreibfehler
1.2	14.10.2015	Bearbeitet	Hohenwarter	uWsgi

2 Hosting

Unser Projektserver ist beim in Deutschland ansässigen Unternehmen Netcup[1] gehostet. Hier haben wir den Root-Server M angemietet(2 Cores Intel®Xeon® E5-26xxV3 (min. 2,3 GHz je Kern), 6 GB DDR4(ECC),120GB SSD, 1 GBit/s)[2] mit dem OS Debian 8 und mit der Kernel Version 3.16.0-4-amd64. Dieser ist unter der IP Adresse **37.120.161.195** erreichbar.

Auf dem Server ist die Domain digitalschoolnotes.com geschaltet. Diese ist ebenfalls bei netcup gemietet.

Folgende Subdomains existieren:

- time.digitalschoolnotes.com Zeitaufzeichnung
- git.digitalschoolnotes.com Repository
- ontime.digitalschoolnotes.com Scrum Tool
- ci.digitalschoolnotes.com CI Tool

3 User

Jedes Projektteam Mitglied hat einen eigenen Unix Account auf dem Projektserver. Der Vorname der Person ist der Benutzername. Das Benutzerpasswort ist von jedem Teammitglied selbst gewählt. Alle Teammitglieder haben sudo rechte. Verantwortlich für die Serververwaltung ist Niklas Hohenwarter. Bei Problemen mit der Anmeldung oder anderem sind diese ihm bekannt zu geben.

4 Mailsystem

Das Projektteam hat einen Email-Verteiler mit der Adresse info@digitalschoolnotes.com. Jedes Teammitglied hat eine E-Mail Adresse nach dem Schema des TGMs. Der Scrummaster ist unter scrummaster@digitalschoolnotes.com erreichbar.

5 SSH

Aus Sicherheitsgründen wurde die Anmeldung mit Passwort verboten und es können hierfür nurnoch SSH Keys verwendet werden.

6 Firewall

Es werden prinzipiell alle eingehenden Ports geschlossen. Ausnahmen sind hier aufzulisten. Bei Änderungswünschen ist der Serveradministrator zu kontaktieren.

Ausnahmen:

- 22 SSH
- 53 DNS
- 80 HTTP
- 443 HTTPS
- 5001-5005 Django Development

6.1 Konfiguration

Die Firewall wird mittels folgenden Befehlen aufgesetzt:

```
# Flush the tables to apply changes
iptables -F

# Default policy to drop 'everything' but our output to internet
iptables -P FORWARD DROP
iptables -P INPUT   DROP
iptables -P OUTPUT  ACCEPT

# Allow established connections (the responses to our outgoing
    traffic)
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Allow local programs that use loopback (Unix sockets)
iptables -A INPUT -s 127.0.0.0/8 -d 127.0.0.0/8 -i lo -j ACCEPT
iptables -A FORWARD -s 127.0.0.0/8 -d 127.0.0.0/8 -i lo -j
    ACCEPT

#Allowed Ports
iptables -A INPUT -p tcp --dport 22 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 80 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 443 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 53 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p udp --dport 53 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 5001 -m state --state NEW -j
    ACCEPT
```

```
iptables -A INPUT -p tcp --dport 5002 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 5003 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 5004 -m state --state NEW -j
    ACCEPT
iptables -A INPUT -p tcp --dport 5005 -m state --state NEW -j
    ACCEPT
```

Die Firewallrules werden beim Reboot automatisch wiederhergestellt. Dies geschieht durch das Paket ***iptables-persistent***. Konfiguration[3]:

```
# Install
sudo apt-get install iptables-persistent

# Save Rules
iptables-save > /etc/iptables/rules.v4
```

7 Bruteforce Prevention

Um den SSH Zugang gegen Brute Force Attacken abzusichern wurde fail2ban installiert. Dieses Paket versucht Bruteforce Attacken zu verhindern. [4]

7.1 Konfiguration

Das Paket wurde mittels ***sudo apt-get install fail2ban*** installiert. Die Standardkonfiguration wurde angepasst, um all unsere Services zu schützen.

8 Webserver

Als Webserver wird nginx verwendet. Aktuell existieren auf dem Server nur Weiterleitungen zu unserem Github Repository, Taiga, der Zeitaufzeichnung und dem CI Tool (Subdomains, siehe oben).

8.1 Konfiguration

Die Konfiguration ist in ***/etc/nginc/sites-available/redirects*** zu finden. Sie enthält folgendes:

```
server {
        listen 80; #Port
        server_name git.digitalschoolnotes.com; # Subdomain

        location / {
            rewrite ^ https://github.com/nhohenwarter-tgm/
                digitalschoolnotes permanent; # zieladresse der
                Weiterleitung
```

```
        }
}

server {
        listen 80;
        server_name ontime.digitalschoolnotes.com;

        location / {
           rewrite ^ tgm.axosoft.com permanent;
        }
}


server {
        listen 80;
        server_name time.digitalschoolnotes.com;

        location / {
           rewrite ^ https://goo.gl/IWrE2j permanent;
        }
}
```

Um die Weiterleitung zu aktivieren muss ein Link nach ***/etc/nginx/sites-enabled*** gesetzt werden. Dies geht mit dem Befehl ***ln -s /etc/nginx/sites-available/redirects /etc/nginx/sites-enabled/redirects***. Dannach muss der Nginx Service neu gestartet werden. Diese geschieht mit dem Befehl ***service nginx reload***.

9 Deployment

9.1 SSL

Um die Daten verschlüsselt übertragen zu können wird ein SSL Zertifikat benötigt. Dieses kaufen wir bei unserem Hoster. Wir haben uns für das Thwawte SSL123 Zertifikat[5] für 1 Jahr entschieden. Bei dem Kauf des Zertifikates muss auf dem Server zuerst eine Zertifikatsanfrage erstellt werden(CSR).

Dies geschieht wie folgt[6]:

```
openssl req -new -newkey rsa:2048 -nodes -sha256 -keyout dsn.key
    -out dsn.csr
Generating a 2048 bit RSA private key
.....................................................................+++

.....+++
writing new private key to 'dsn.key'
-----
You are about to be asked to enter information that will be
   incorporated
into your certificate request.
```

```
What you are about to enter is what is called a Distinguished
   Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:AT
State or Province Name (full name) [Some-State]:Wien
Locality Name (eg, city) []:Wien
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
   DigitalSchoolNotes
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
   digitalschoolnotes.com
Email Address []:root@digitalschoolnotes.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

Nach dem beantworten der Fragen wird das CSR generiert. Dieses muss dann beim Kauf angegeben werden. Dannach muss das Zertifikat bezahlt werden. Nun bekommt man das Zertifikat per Email.

Nun muss das Zertifikat auf Nginx eingerichtet werden[7]: Man benötigt das Zertifikat, das öffentliche Zertifikat von Thawte und den Private Key. Diese müssen nun zusammengefügt werden.

```
cat digitalschoonotes.com.crt intermediate.crt >
   digitalschoolnotes.com.chain.crt
```

Nun muss Nginx konfiguriert werden.

```
#/etc/nginx/sites-available/default
server {
        listen 80;
        listen 443 ssl;

        root /var/www/html;

        # Add index.php to the list if you are using PHP
        index index.html index.htm index.nginx-debian.html;

        server_name digitalschoolnotes.com;
        ssl_certificate /home/niklas/ssl/digitalschoolnotes.com.
           chained.crt;
        ssl_certificate_key /home/niklas/ssl/digitalschoolnotes.
           com.key;

        ssl_protocols TLSv1.2;
```

```
        ssl_prefer_server_ciphers on;
        ssl_ciphers AES256+EECDH:AES256+EDH:!aNULL;
        ssl_verify_depth 3;

        if ($ssl_protocol = "") {
            rewrite ^   https://$server_name$request_uri?
               permanent;
        }

        location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a
                   404.
                try_files $uri $uri/ =404;
        }
}
```

Es werden nun alle HTTP Anfragen nach HTTPS umgeleitet.

9.2 Gunicorn

Wir haben uns dazu entschieden in Entwicklung und Produktiveinsatz Django mit gunicorn zu deployen. Nginx beantwortet alle statischen Anfragen. Fall etwas von Django abgefragt werden muss, leitet Nginx die Anfrage an Djano weiter. Die Konfiguration läuft wie folgt ab[8]:
Als erstes muss gunicorn installiert werden:

```
sudo pip3 install gunicorn
```

Nun verwenden wir Supervisor um gunicorn[9] auszuführen. Supervisor kann das ganze dann über einen Kommander starten und stoppen. Zunächst erstellen wir ein Konfigurationsfile:

```
sudo vim /etc/supervisor/conf.d/dsnselina.conf

[program:dsnselina]
directory=/home/selina/dsn
user=selina
command=gunicorn --bind 0.0.0.0:8001 dsn.wsgi:application
autostart=true
autorestart=true
```

Nun können wir über den Kommander das Programm dsnselina starten oder beenden.

```
sudo supervisorctl

start dsnselina #Starten
stop dsnselina #Stoppen
status #Status aller Programme
```

Zuletzt fehlt noch die Konfiguration von Nginx. Hierzu legen wir ein neues Konfigurationsfile an:

```
cd /etc/nginx/sites-available
```

```
cp default selina
vim selina

server {
        listen 5001;

        root /home/selina/dsn/dsn/static;

        # Add index.php to the list if you are using PHP
        index index.html index.htm index.nginx-debian.html;

        server_name digitalschoolnotes.com;

        location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a
                   404.
                try_files $uri $uri/ =404;
        }

        location /api/ {
                proxy_pass  http://127.0.0.1:8001;
                proxy_set_header Host $host;
                proxy_set_header X-Real-IP $remote_addr;
                proxy_set_header X-Forwarded-For
                   $proxy_add_x_forwarded_for;
        }
}

ln -s selina ../sites-enabled/selina
```

10 Literatur

[1] Netcup, "netcup - webhosting, vserver, servermanagement." http://netcup.de/. zuletzt besucht: 09.09.2015.

[2] Netcup, "netcup - webhosting, vserver, servermanagement." https://www.netcup.de/vserver/#features. zuletzt besucht: 09.09.2015.

[3] T. Krenn, "Iptables firewall regeln dauerhaft speichern." https://www.thomas-krenn.com/de/wiki/Iptables_Firewall_Regeln_dauerhaft_speichern. zuletzt besucht: 21.09.2015.

[4] T. Krenn, "Ssh login unter debian mit fail2ban absichern." https://www.thomas-krenn.com/de/wiki/SSH_Login_unter_Debian_mit_fail2ban_absichern. zuletzt besucht: 09.09.2015.

[5] Netcup, "Thawte ssl123 zertifikat 1 jahr." https://www.netcup.de/bestellen/produkt.php?produkt=263. zuletzt besucht: 12.10.2015.

[6] N. Wiki, "Key und csr erstellen." http://www.netcup-wiki.de/wiki/Key_und_CSR_erstellen. zuletzt besucht: 12.10.2015.

[7] M. Anicas, "How to install an ssl certificate from a commercial certificate authority." https://www.digitalocean.com/community/tutorials/how-to-install-an-ssl-certificate-from-a-commercial-certificate-authority. zuletzt besucht: 12.10.2015.

[8] J. Ellingwood, "How to set up django with postgres, nginx, and gunicorn on ubuntu 14.04." https://www.digitalocean.com/community/tutorials/how-to-set-up-django-with-postgres-nginx-and-gunicorn-on-ubuntu-14-04. zuletzt besucht: 18.10.2015.

[9] DigitalOcean, "How to install and manage supervisor on ubuntu and debian vps." https://www.digitalocean.com/community/tutorials/how-to-install-and-manage-supervisor-on-ubuntu-and-debian-vps. zuletzt besucht: 18.10.2015.