

Praktische Prüfungen im CCNA

Final Skills Exam mit einem Instruktoren-Server

Nationaler Akademietag 2025

Tobias Heine | 23.05.2025 | @toheine@social.tchncs.de

Ablauf

- Warum überhaupt der ganze Aufwand?
- Vorstellung des Konzepts
- Kurze Demo
- Derzeitige Grenzen

Ausgangssituation - Lehrgangsdesign

- CCNA-Lehrgänge in der Lehrkräftefortbildung BW finden in 6x2,5-tägigen Präsenzphasen statt.
- Dazu kommen etliche Stunden der Vorbereitung durch die Teilnehmerinnen (TNs) daheim.
- Pro CCNA-Kurs (ITN, SRWE, ENSA) zwei Präsenztermine.
- Am Ende von Lehrgangs-Teil 2, 4 und 6 finden für jeden Teilnehmer:
 - 1x Final Exam
 - 1x Final Skills Exam (Equipment oder PTSA)
- Erfolgreiches Bestehen führt zur Lehrqualifikation für den jeweiligen Kurs.

Ausgangssituation

- ITN und SRWE
 - an Geräten
 - Vorbereitete Aufgaben verfügbar
 - Für uns als Orientierung für eigene Aufgabenstellung
- ENSA
 - mit Packet-Tracer (verfügbares PTSA)

The screenshot shows the Cisco Networking Academy interface for the course CISI25-CCNAv7-ItN. The 'Resources' tab is active, displaying a search bar and a list of course resources. A red arrow points from the text 'Für uns als Orientierung für eigene Aufgabenstellung' to the 'CCNAv7 ITN Instructor Skills Assessment' resource.

Course Outline	Resources
<input type="text" value="Search Resources"/>	
Course Resources	
	CCNAv7 ITN Instructor Skills Assessment

Ausgangssituation

- Mehrwert von Prüfungen an echten Geräten, weil ...
 - ... TNs sich von Beginn an mit echten Geräten in der Präsenzzeit vorbereiten (PacketTracer eher zu Hause).
 - ... dadurch für den Unterricht mehr Troubleshooting-KnowHow vermittelt wird (defekte Kabel, Handling von Kabelsalat, Stromzufuhr, etc.).
 - ... wenig bis keine Cheating-Möglichkeiten mit Braindumps möglich sind. (Ohne Vorwurf!!! Ohne Unterstellung!!!)

Ausgangssituation

Das sieht dann etwas so aus:

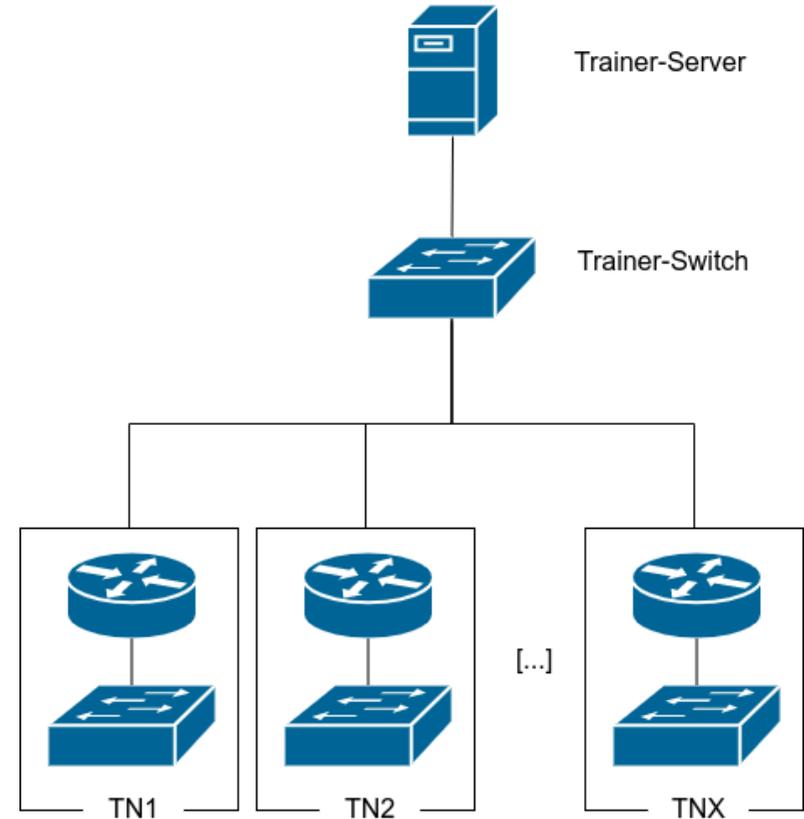


Probleme bei praktischen Prüfungen (früher)

- Durchführung als letzter Programmpunkt eines Präsenztermins.
- TNs auf dem Sprung nach Hause.
- Anschlussveranstaltung in der Einrichtung erhöht Druck zur schnellen Korrektur an Geräten.
- Korrektur an Geräten zeitaufwendig.
- Doofes Szenario: TN vertippt sich bei Vergabe von Passwörtern und ist nicht bei Korrektur verfügbar.

Mögliche Lösung

- Die einzelnen TN-Umgebungen an eine zentrale Infrastruktur anschließen und
- Server bereit stellen der Dienste vorhält (z. B. TFTP-Server)
- TN spielen am Ende der Prüfung ihre Konfigurationen auf den TFTP-Server



Erste Umsetzung

- Kleinen PC/NUC für den Trainer-Server im Keller gesucht und gefunden.
- Auf dem Ding eine Linux-Distribution (Debian) installiert.
- Entsprechend viele Sub-Interfaces auf dem Server bereit gestellt.
- TFTP-Service (`atftp`) installiert, auf dem die TNs ihre Lösungen hochladen.
- Dafür gesorgt, dass sich die TNs in den Lösungen NICHT überschreiben:
→ durch Vergabe von Nick-Names in Dateinamen.
- Individuelle Aufgabenstellung durch Serienbrief.

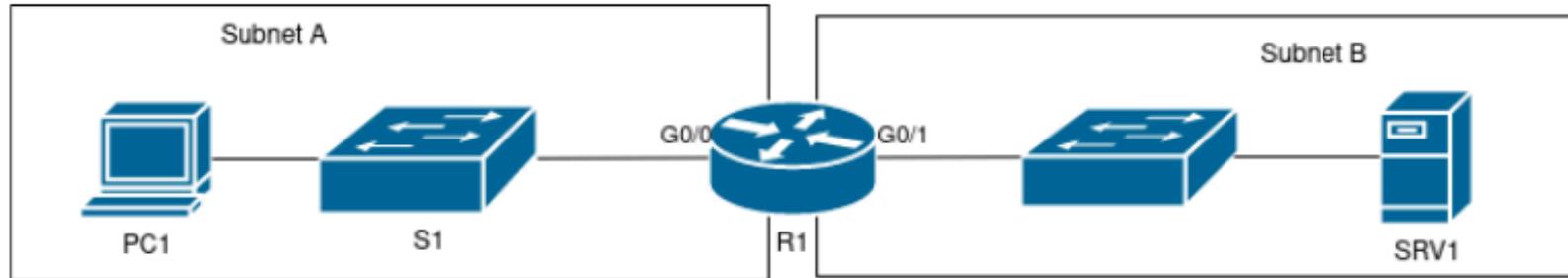
Auszug aus Aufgabenstellung (ITN)

Individuelle Nicknames, IDs und Ports auf Instruktoren-Switch

Time: 100 minutes / max. 78 point (+ 2 extra points for an experimental task)

Your ID: 17	Important: Connect R1 G0/0 to the Subnet-B-Switch on Port FE0/1
Name: _____	Nickname: beef

Topology



Auszug aus Aufgabenstellung (ITN)

Individuelles IP-Adress-Schema. Via Serienbrief pro Student im individuellen Subnet:

Device	IP address	Subnet Mask	Gateway
PC1	10.17.1.1	255.255.255.0	10.17.1.254
S1	10.17.1.253	255.255.255.0	10.17.1.254
R1-G0/0	10.17.1.254	255.255.255.0	N/A
R1-G0/1	10.17.0.2	255.255.255.252	N/A
SRV1	10.17.0.1	255.255.255.252	N/A

ID hier im zweiten Oktett

Auszug aus Aufgabenstellung (ITN)

Individuelle Dateinamen um Überschreibungen zu vermeiden
(Security by obscurity ;-)

Save the current configuration of S1 to the TFTP-Server 10.17.0.1 and use the file-name: beef-s1-config	copy run tftp: [Enter] <i>use assistent for more entries</i>	1 extra point
Save the current configuration of R1 to the TFTP-Server 10.17.0.1 and use the file-name: beef-r1-config	copy run tftp: [Enter] <i>use assistent for more entries</i>	1 extra point

note: saving the current configuration is in “first time use” during this exam. Technical or organizational problems are possible. Every student who accomplish this task, will earn up to 2 extra points

Feinarbeiten

- Grundlage für Serienbrief und automatisierter Bereitstellung der Infrastruktur ist die Datei `subnets.csv`.
- Um Client-Konfiguration zu prüfen, wird ein Webserver bereit gestellt der pro TN und IP-Protokoll eine individuelle Webseite ausliefert.
(max. 23x2 `nginx`-Serverblöcke)
- Jede dieser Webseiten liefert eine individuelle Lösung aus.
- Hat ein TN die korrekten Lösungen auf seiner Prüfung notiert, ist eine Überprüfung des Clients nicht mehr notwendig.
- Zusätzliche notwendig: DNS-Server (`bind9`)

Auszug aus Aufgabenstellung

Abfrage der individuellen IPv4-Lösung

Open a browser and visit the website **http://4web.beef.ccna-lab.com**. Note the Keyword from the given website in the next line:

IPv4 Website

Hallo,

Du hast die IPv4-Konfiguration auf Deinem PC korrekt durchgeführt.
Bitte notiere folgende "Lösung" in der entsprechende Zeile Deiner Prüfung:

beef broth

Auszug aus Aufgabenstellung

Abfrage der individuellen IPv6-Lösung

Open a browser and visit the website **http://6web.beef.ccna-lab.com**. Note the Keyword from the given website in the next line:

IPv6 Website

Hallo,

Du hast die IPv6-Konfiguration auf Deinem PC korrekt durchgeführt.
Bitte notiere folgende "Lösung" in der entsprechende Zeile Deiner Prüfung:

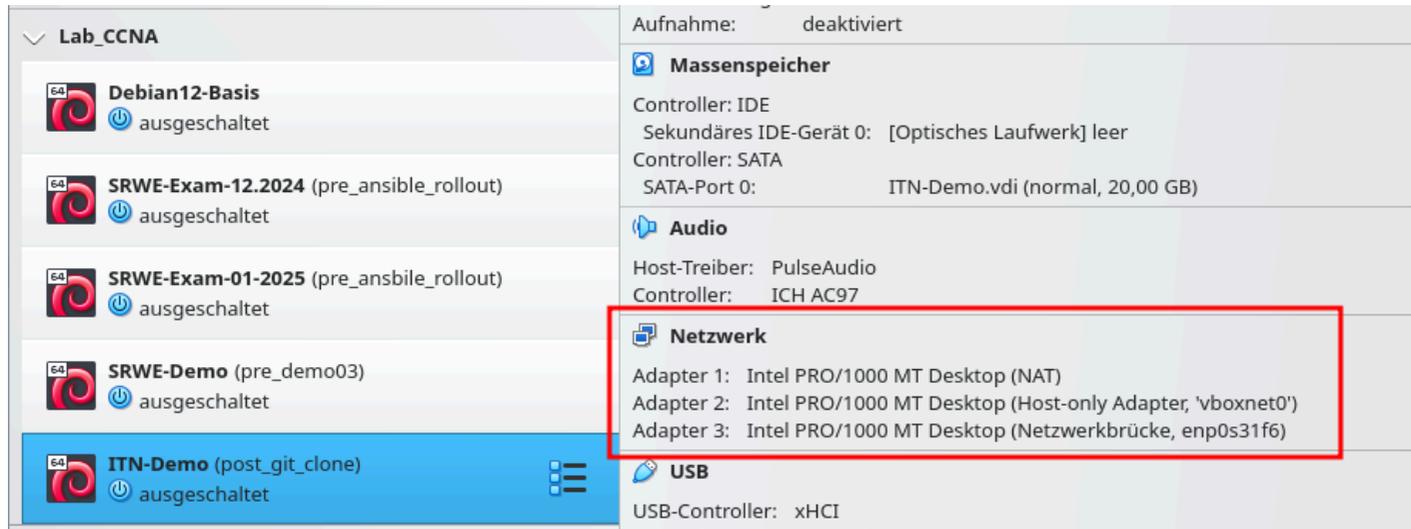
spiced beef broth

Fazit vor Demo

- TNs die die Aufgaben weitgehend lösen, können nach ...
 - ... Abgabe der Konfigdateien und
 - ... zwei kleinen Checks (Konfig auf TFTP-Server vorhanden, SSH-Login möglich) durch die Instruktoren
 - ... ihr Lab direkt abbauen.
- Korrektur der Konfigs am Server deutlich schneller.
- Nur noch wenige TNs müssen am Gerät korrigiert werden.
- Letzter Schritt: Den ganzen Kladderadatsch in **Ansible** gegossen:
<https://codeberg.org/toheine/ccna-final-skills-exam-infra>

Demo - Schritt 1: Server bereit stellen

- Kleine Kiste mit Debian 12 und zwei Netzwerk-Schnittstellen oder
- Virtuelle Maschine mit Debian 12 auf Instruktoren-Notebook
(Min. zwei Netzwerk-Interfaces)



Demo - Schritt 2: Doku abarbeiten

- Git-Repo: <https://codeberg.org/toheine/ccna-final-skills-exam-infra>

☐ README.md

Einrichtung eines Debian-Servers für die praktische CCNA-Prüfung

Vorbereitung

- Einen Server mit Debian 12 bereitstellen (kann eine VM sein)
- Server sollte 2 oder mehr Netzwerk-Interfaces haben (1x für Exam-Lab + 1x regulär [+ z. B. Host-Only-Adapter bei VM])
 - 1x NAT oder Bridged: darüber werden Updates und das vorbereitete git-Repo bezogen. IP-Adressen kommen aus dem produktiven Netz via DHCP
 - 1x Bridged: wird das Labor-Netz. Die IP-Adressen werden via ansible konfiguriert.
 - evtl. 1x Host-Only: kann bei einer VM sinnvoll sein um sich zu Beginn von localhost direkt auf die VM zu verbinden.
- Erste Basis-Tools installieren und Repo klonen:

```
sudo apt install ansible git
git clone https://codeberg.org/toheine/ccna-final-skills-exam-infra.git
```

Demo - subnets.csv

Grundlage für Serienbrief und ansible-vars

main [cna-final-skills-exam-infra / subnets.csv](#)



Tobias Heine

61dd9f84b7

Initial commit

vor 17 Stunden

1,8 KiB



Originalformat

Permalink

Blame

Verlauf



	id	switchport	ipv6-subnet-a	ipv6-subnet-b	nickname	switch-config	router-config	ipv4code	ipv6code
2	17	FE0/1	11	31	beef	beef-s1-config	beef-r1-config	beef broth	spiced beef broth
3	18	FE0/2	12	32	berg	berg-s1-config	berg-r1-config	berg is busy	berg is very busy
4	19	FE0/3	13	33	bird	bird-s1-config	bird-r1-config	raven	eagle
5	20	FE0/4	14	34	boat	boat-s1-config	boat-r1-config	sailboat	sailing ship

Demo - Netzwerkonfiguration

Pro Teilnehmer eigenes Subinterface

- Erfolgt regulär in der `/etc/network/interfaces`.
- Darin Include-Anweisung für `/etc/network/interfaces.d/` enthalten.
- Dort liegen die einzelnen Konfigurationen:

```
root@trainserver:~# ls -l /etc/network/interfaces.d/  
insgesamt 96  
-rw-r--r-- 1 root root 193 21. Mai 17:18 enp0s9.100  
-rw-r--r-- 1 root root 538 21. Mai 17:18 enp0s9.17  
-rw-r--r-- 1 root root 538 21. Mai 17:18 enp0s9.18  
-rw-r--r-- 1 root root 538 21. Mai 17:18 enp0s9.19  
-rw-r--r-- 1 root root 538 21. Mai 17:18 enp0s9.20  
[ ... ]
```

Demo - Netzwerkonfiguration

Subinterface konfiguriert für IPv4 und IPv6

```
# _____  
# Interface vlan 17  
# _____  
auto enp0s9.17  
iface enp0s9.17 inet static  
    address 10.17.0.1  
    netmask 255.255.255.0  
    up ip route add 10.17.0.0/16 via 10.17.0.2  
iface enp0s9.17 inet6 static  
    address FE80::CAFE  
    netmask 64  
iface enp0s9.17 inet6 static  
    address 2001:DB8:17:1::1  
    netmask 64  
pre-up echo 0 > /proc/sys/net/ipv6/conf/enp0s9.17/accept_dad  
up ip -6 route add 2001:DB8:17::/48 via 2001:DB8:17:1::2
```

Demo - Webserver mit nginx

Pro Teilnehmer ein eigener VirtualHost-Block

```
root@trainserver:~# ls -l /etc/nginx/sites-available/  
insgesamt 96  
-rw-r--r-- 1 root root 542 21. Mai 17:25 beef  
-rw-r--r-- 1 root root 542 21. Mai 17:25 berg  
-rw-r--r-- 1 root root 542 21. Mai 17:25 bird  
[ ... ]  
  
root@trainserver:~# cat /etc/nginx/sites-available/beef  
server {  
    listen 10.17.0.1:80;  
    root /var/www/4web.beef;  
    [ ... ]  
  
server {  
    listen [2001:DB8:17:1::1]:80;  
    root /var/www/6web.beef;  
    [ ... ]
```

Demo - Webserver mit nginx

Webseiten in `/var/www`

Seiten mit den Lösungen hinterlegt:

```
root@trainserver:~# ls -l /var/www/ | egrep -i "(beef|berg|bird)"
drwxr-xr-x 2 root root 4096 21. Mai 17:25 4web.beef
drwxr-xr-x 2 root root 4096 21. Mai 17:25 4web.berg
drwxr-xr-x 2 root root 4096 21. Mai 17:25 4web.bird
drwxr-xr-x 2 root root 4096 21. Mai 17:25 6web.beef
drwxr-xr-x 2 root root 4096 21. Mai 17:25 6web.berg
drwxr-xr-x 2 root root 4096 21. Mai 17:26 6web.bird

root@trainserver:~# cat /var/www/4web.beef/index.html | grep -A4 Hallo
<p>Hallo,</p>
<p>Du hast die IPv4-Konfiguration auf Deinem PC korrekt durchgeführt.
<br/>Bitte notiere folgende "Lösung" in der entsprechende Zeile Deiner Prüfung:</p>
<p><b>beef broth</b></p>
</main>
```

Demo - DNS-Server mit bind9

Abfrage jeweils nur aus bestimmter Zone erlaubt

```
root@trainserver:~# cat /etc/bind/named.conf.local | grep -A 15 "zone beef"
// zone beef
zone "beef.ccna-lab.com" IN {
    type master;
    file "db.beef.ccna-lab.com.zone";
    allow-query {
        10.17.0.0/16;
        2001:DB8:17:1::0/64;
        2001:DB8:17:2::0/64;
        2001:DB8:17:3::0/64;
        2001:DB8:17:4::0/64;
        2001:DB8:17:6::0/64;
        127.0.0.1;
        ::1;
    };
};
```

Demo - DNS-Server mit bind9

DNS-Zone löst korrekt auf:

```
root@trainserver:~# cat /var/cache/bind/db.beef.ccna-lab.com.zone
$TTL 300

@ IN SOA srv1.beef.ccna-lab.com. root.localhost. (
    2022120603 ; Serial
    86400 ; Refresh
    7200 ; Retry
    604800 ; Expire
    172800 ; Minimum
)

@           NS      srv1.beef.ccna-lab.com.

srv1       A        10.17.0.1
4web       A        10.17.0.1
6web       AAAA     2001:db8:17:1::1
```

Derzeitige Grenzen / offene Baustellen

- System kann auf jeden Fall noch optimiert werden.
- Eigene Arbeitszeit allerdings nur beschränkt verfügbar.
- Überschreibungen auf TFTP-Instanz derzeit noch möglich, sofern TNs im Nickname abweichen (würden).
- Theoretisch möglich: "bessere" Konfigs von Sitznachbarn herunter zu laden.
- Automatisierte Korrektur findet noch nicht statt. (Prio 1)
- VLAN.dat derzeit nur nach Login vom Lehrer-Gerät einsehbar.
- SSH-Keylänge derzeit nur bei Login vom Instruktoren-Server einsehbar.

Fragen, Anregungen, Kommentare (???)

- Gerne hier direkt oder später in den Fluren,
- als Issue im angegebenen Git-Repo oder
- über folgende Kanäle
 - Mail: mail@toheine.net
 - PM auf Mastodon: @toheine@social.tchncs.de

Danke für's Zuhören!!!