

Bash Kurs Übung: Dienst Monitor. dienstmon

Ziel: dienstmon zeigt die offen Dienst-Port von einem Host.

Teilnehmer sind frei zwieschen verschiedene Folgende Script-Typ zu whälen :

Voraussetzungen: nmap Packet muss installiert sien.

A) Zeigt (dynamische)welche spezifische Dienst sind offen in einem host:

Syntax: dienstmon hostname dienstlist
z.B. dienstmon laptop.linux.site 21 22 25 80 110
oder dienstmon laptop.linux.site 21,22,25,80,110

Ergebnis:

Ein xterm erscheint mit die Folgende Zeile:(Nur eine Beispiel)
(ROT=Deaktiviert GRÜN=Aktiviert)

```
laptop.linux.site FTP SSH SMTP HTTP POP3
```

B) Zeigt alle offene Dienst-Port von einem Host.

Syntax: dienstmon hostname
z.B. dienstmon laptop.linux.site

Ergebnis:

xterm erscheint mit die Folgende Zeile:(Nur eine Beispiel)

```
laptop.linux.site FTP SSH SMTP HTTP POP3 HTTPS SWAT
```

**C) Zeigt bestimmte offene Dienst-Ports für alle rechner is Lokales Netzwerk:
(Für Vortgeschriene)**

Syntax: dienstmon 21,22,25,80,110,901

Ergebnis:

xterm erscheint mit die Folgende Zeilen:(Nur eine Beispiel)

```
linpc01.linux.site    FTP SSH SMTP HTTP POP3 SWAT
linpc02.linux.site    FTP SSH SMTP HTTP POP3 SWAT
linpc03.linux.site    FTP SSH SMTP HTTP POP3 SWAT
linpc04.linux.site    FTP SSH SMTP HTTP POP3 SWAT
linpc05.linux.site    FTP SSH SMTP HTTP POP3 SWAT
linpc06.linux.site    FTP SSH SMTP HTTP POP3 SWAT
.....u.s.w.
```

oder(vielleich besser)

	<u>FTP</u>	<u>SSH</u>	<u>SMTP</u>	<u>HTTP</u>	<u>POP3</u>	<u>SWAT</u>
linpc01.linux.site	○	○	○	○	○	○
linpc02.linux.site		○	○	○	○	○
linpc03.linux.site	○	○		○		○
linpc04.linux.site		○	○		○	○
linpc05.linux.site	○	○		○		○
linpc06.linux.site	○	○		○		○
.....u.s.w.						

Tips:

- **Wieder schreiben auf dass selbe Zeile in xterm:**

```
echo -ne "Hallo Martin"
echo -ne "\rChuss"
```

Ergebnis: 'Hallo Martin' ist geschrieben,
dann der Cursor geht am anfang die Zeile(\r)
und überschreibt Hallo mit Chuss .

- **Terminal wieder sauber machen**

Befehl: `clear`

- **Befehl um eine eizel Dienst-Port zu überprüfen:**

(z.B. Port 80 auf `laptop.linux.site`)

```
nmap -n --host_timeout 210 --max_rtt_timeout 210 -p 80 -P0 \
laptop.linux.site | grep open &>/dev/null
```

Die 'exit code' von diese Befehl ist:

0 wenn die Port 80 auf `laptop.linux.site` host ist offen

1 wenn die Port 80 auf `laptop.linux.site` host ist NICHT offen

Beispiel:

```
if (nmap -n --host_timeout 210 --max_rtt_timeout 210 -p 80 -P0 \
laptop.linux.site | grep open &>/dev/null); then
echo "HTTP Port auf laptop.linux.site ist OFFEN"
else
echo "HTTP Port auf laptop.linux.site ist GESCHLOSSEN"
fi
```

- **Funktion(`show`) um eine wort mit Leerzeichen bis bestimmte Länge ergänzen und mit Schrift Farbe und Hintergrund Farbe auf STDOUT zu zeigen. Die Normale cheichen sind in Farbe aber die Leerzeichen sind in Weiss.**

z.B

```
show "Hallo" 8 ROT GELB-H ; show "Julie" 10 GELB SCHWARTZ-H
```

Ergebnis: **Hallo** **Julie**

```

#--- Function to display a word ($1) and fill the rest with spaces
#--- until the length of the word is according to parameter $2
#--- Character Color according to $3
#--- Background Color according to $4
#--- Syntax: eg. show SSH 8 ROT GELBE-H
#--- Would display the word SSH in red on yellow
#--- and add 5 spaces at the end of it.(8-3=5)
show ()
{
    #-----Effekte:
    FETT="\033[1m"
    UNTERSTRICHT="\033[4m"
    FLACKER="\033[5m"
    UNSICHBAR="\033[8m"

    #-----Schrift Farbe:
    ROT="\033[31m"
    GRUEN="\033[32m"
    GELB="\033[33m"
    BLAU="\033[34m"
    VIOLET="\033[35m"
    HELLBLAU="\033[36m"
    WEISS="\033[37m"
    SCHWARTZ="\033[39m"

    #-----Hintergrund Farbe:
    SCHWARTZ_H="\033[40m"
    ROT_H="\033[41m"
    GRUEN_H="\033[42m"
    GELB_H="\033[43m"
    BLAU_H="\033[44m"
    VIOLET_H="\033[45m"
    HELLBLAU_H="\033[46m"
    WEISS_H="\033[47m"

    #Schwartz auf Weiss, Effekt sind ausgeschaltet
    NORMALE="\033[m"

    #---Hintergrund nimmt die Schrift Farbe
    #---und die Schritten nehmen eine sichtbare Farbe)
    REVERSE="\033[7m"

    #----- Leerzeichen vor und nach den Wort weg nehmen
    word=$(echo -n $1 | sed -e 's/^ */g' -e 's/ */g') ;
    spaces="";
    let sp=$2-$(echo -n $word | wc -m);
    echo -ne $(eval echo \$$3)$(eval echo \$$4)$word$NORMALE$(echo
-n $spaces | cut -b $sp)
}

```