

# SLAC 2024

06.-08. Mai 2024 | Berlin

[www.slac-2024.de](http://www.slac-2024.de)

7. Mai 2024 / Mirko Ludeke

# Mosh & screen

# Inhalt



- Was ist screen?
- Was ist mosh?
- Vor- und Nachteile

# Was ist screen?



- Terminal-Multiplexer (“Terminal im Terminal”)
- Fenstermanager zur Verwendung mit textbasierten Eingabefenstern, um mehrere virtuelle Konsolensitzungen zu erzeugen und zu verwalten

```
Screen key bindings, page 1 of 3.

Command key: ^A  Literal ^A:  a

break      ^B b          license      ,          removebuf   =
clear      C          lockscreen  ^X x          reset       Z
colon      :          log         H          screen      ^C c
copy       ^[ [          login       L          select      '
detach     ^D d          meta        a          silence     _
digraph    ^V          monitor     M          split       S
displays   *          next        ^@ ^N sp n    suspend     ^Z z
dumptermcap .        number      N          time        ^T t
fit        F          only        Q          title       A
flow       ^F f          other       ^A          vbell      ^G
focus     ^I          pow_break   B          version     v
hardcopy   h          pow_detach  D          width       W
help       ?          prev        ^H ^P p ^?   windows     ^W w
history    { }          quit        ^\          wrap        ^R r
info       i          readbuf     <          writebuf    >

[Press Space for next page; Return to end.]
```

# ssh-Screen



- Mit der folgenden Erweiterung am Ende der Datei `.bash_profile` wird automatisch mit einer screen-Sitzung verbunden, wenn die Verbindung per ssh erfolgt. Bestehende Verbindungen zu der Session werden ggf. getrennt:

```
if [ "$TERM" != "screen" ] && [ "$SSH_CONNECTION" != "" ]; then  
    /usr/bin/screen -S sshscreen -d -R && exit  
fi
```

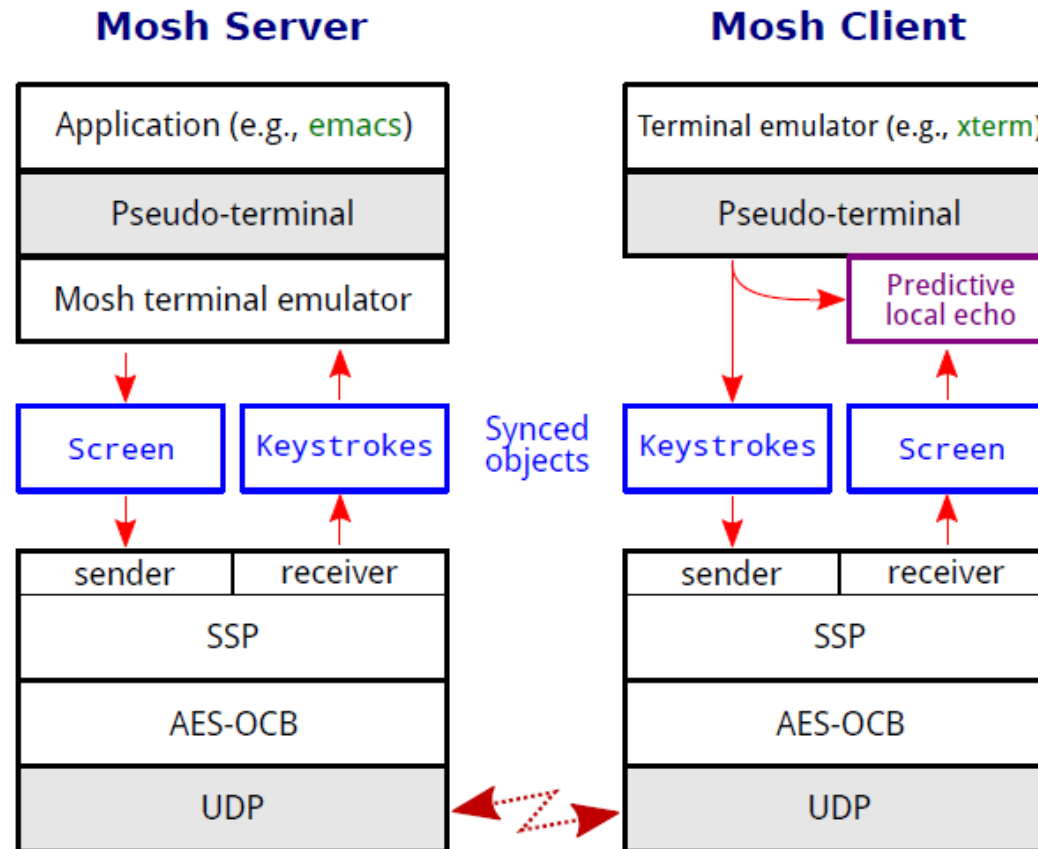
- Wird screen beendet oder getrennt, wird auch die SSH-Verbindung getrennt – so ist also nur eine SSH-Verbindung möglich. Das kann umgangen werden, indem man `&& exit` weglässt.

# Was ist mosh?



- mosh steht für mobil shell
- plattformunabhängig
- setzt da an, wo SSH schwächelt
- kein Ersatz für SSH, aber irgendwie schon...
- Nutzt State Synchronisation Protocol (SSP) => UDP statt TCP (!)
- Objekt Diff/Patch statt Screen Übertragung
- Verschlüsselung AES OCB3 (Offset codebook mode)

# Mosh - Funktionsweise



# Mosh - Vergleich zu SSH 1/2



No.	Time	Source	Destination	Protocol	Length	Data	Info
1	0.00000000	172.16.51.46	172.16.6.72	SSHV2	107		Client: Protocol (SSH-2.0-openssh_6.6.1p1 Ubuntu-2ubuntu2)
2	0.00975200	172.16.6.72	172.16.51.46	SSHV2	107		Server: Protocol (SSH-2.0-openssh_5.9p1 Debian-5ubuntu1.4)
3	0.01068200	172.16.6.72	172.16.51.46	SSHV2	1050		Server: Key Exchange Init
4	0.01074400	172.16.51.46	172.16.6.72	TCP	1514		[TCP segment of a reassembled PDU]
5	0.01075900	172.16.51.46	172.16.6.72	SSHV2	586		Client: Key Exchange Init
6	0.01135300	172.16.51.46	172.16.6.72	SSHV2	146		Client: Diffie-Hellman Key Exchange Init
7	0.02024700	172.16.6.72	172.16.51.46	SSHV2	378		Server: Diffie-Hellman Key Exchange Reply, New Keys
8	0.02328600	172.16.51.46	172.16.6.72	SSHV2	82		Client: New Keys
9	0.06070400	172.16.51.46	172.16.6.72	SSHV2	114		Client: Encrypted packet (len=48)
10	0.06107200	172.16.6.72	172.16.51.46	SSHV2	114		Server: Encrypted packet (len=48)
11	0.06193800	172.16.51.46	172.16.6.72	SSHV2	130		Client: Encrypted packet (len=64)
12	0.06501300	172.16.6.72	172.16.51.46	SSHV2	194		Server: Encrypted packet (len=128)
13	8.46799600	172.16.51.46	172.16.6.72	SSHV2	210		Client: Encrypted packet (len=144)
14	8.50602700	172.16.6.72	172.16.51.46	SSHV2	98		Server: Encrypted packet (len=32)
15	8.50639700	172.16.51.46	172.16.6.72	SSHV2	194		Client: Encrypted packet (len=128)
16	8.66805400	172.16.6.72	172.16.51.46	SSHV2	114		Server: Encrypted packet (len=48)
17	8.66836100	172.16.51.46	172.16.6.72	SSHV2	514		Client: Encrypted packet (len=448)
18	8.66990600	172.16.6.72	172.16.51.46	SSHV2	178		Server: Encrypted packet (len=112)
19	8.67387400	172.16.6.72	172.16.51.46	SSHV2	354		Server: Encrypted packet (len=288)
20	9.13707300	172.16.6.72	172.16.51.46	SSHV2	162		Server: Encrypted packet (len=96)
21	10.8827320	172.16.51.46	172.16.6.72	SSHV2	114		Client: Encrypted packet (len=48)
22	10.8832740	172.16.6.72	172.16.51.46	SSHV2	114		Server: Encrypted packet (len=48)
23	11.0163630	172.16.51.46	172.16.6.72	SSHV2	114		Client: Encrypted packet (len=48)
24	11.0168620	172.16.6.72	172.16.51.46	SSHV2	114		Server: Encrypted packet (len=48)



# Mosh - Vergleich zu SSH 2/2



31	2.99323900	172.16.6.72	172.16.51.46	SSHv2	178	Server: Encrypted packet (len=112)
32	2.99366300	172.16.51.46	172.16.6.72	SSHv2	98	Client: Encrypted packet (len=32)
33	3.00243300	172.16.6.72	172.16.51.46	SSHv2	146	Server: Encrypted packet (len=80)
34	3.00248800	172.16.6.72	172.16.51.46	SSHv2	114	Server: Encrypted packet (len=48)
35	3.00256700	172.16.51.46	172.16.6.72	TCP	66	49785->22 [ACK] Seq=3050 Ack=1834 win=36096 Len=0 Tsval=907540 TSecr=162127
36	3.00400100	172.16.6.72	172.16.51.46	SSHv2	498	Server: Encrypted packet (len=432)
37	3.00405700	172.16.6.72	172.16.51.46	SSHv2	130	Server: Encrypted packet (len=64)
38	3.00412500	172.16.51.46	172.16.6.72	TCP	66	49785->22 [ACK] Seq=3050 Ack=2330 win=38016 Len=0 Tsval=907540 TSecr=162127
39	3.00454100	172.16.51.46	172.16.6.72	SSHv2	162	Client: Encrypted packet (len=96)
40	3.00469900	172.16.51.46	172.16.6.72	TCP	66	49785->22 [FIN, ACK] Seq=3146 Ack=2330 win=38016 Len=0 Tsval=907540 TSecr=162127
41	3.00621100	172.16.6.72	172.16.51.46	TCP	66	22->49785 [FIN, ACK] Seq=2330 Ack=3147 win=26064 Len=0 Tsval=16212718 TSecr=907540
42	3.00626800	172.16.51.46	172.16.6.72	TCP	66	49785->22 [ACK] Seq=3147 Ack=2331 win=38016 Len=0 Tsval=907541 TSecr=162127
43	3.01767000	172.16.51.46	172.16.6.72	UDP	111	00000000000000000077316173d Source port: 53749 Destination port: 60002
44	3.02409100	172.16.6.72	172.16.51.46	UDP	369	800000000000000000607edb7d8 Source port: 60002 Destination port: 53749
45	3.12530500	172.16.51.46	172.16.6.72	UDP	104	0000000000000000001f72ffca28 Source port: 53749 Destination port: 60002
46	3.13437300	172.16.6.72	172.16.51.46	UDP	118	80000000000000000019f98b3229 Source port: 60002 Destination port: 53749
47	3.23530500	172.16.51.46	172.16.6.72	UDP	108	000000000000000000250bba0cca Source port: 53749 Destination port: 60002
48	3.48483900	172.16.6.72	172.16.51.46	UDP	151	8000000000000000002a65831888 Source port: 60002 Destination port: 53749
49	3.58617200	172.16.51.46	172.16.6.72	UDP	115	00000000000000000036d57a67c5 Source port: 53749 Destination port: 60002
50	6.48837100	172.16.6.72	172.16.51.46	UDP	108	8000000000000000003cc38328c2 Source port: 60002 Destination port: 53749
51	6.58747500	172.16.51.46	172.16.6.72	UDP	119	00000000000000000044441c766f Source port: 53749 Destination port: 60002
52	9.49170200	172.16.6.72	172.16.51.46	UDP	105	8000000000000000004932c52186 Source port: 60002 Destination port: 53749
53	9.58780200	172.16.51.46	172.16.6.72	UDP	111	000000000000000000529bca2e7b Source port: 53749 Destination port: 60002

# Mosh - Disconnect & Roaming



No.	Time	Source	Destination	Protocol	Length	Data	Info
587	71.138061	172.16.6.15	172.16.6.11	UDP	123	00000000000000d8e1f98d7f3	Source port: 36985 Destination port: 60001
588	74.070774	172.16.6.15	172.16.6.11	UDP	108	80000000000000137651463cc8	Source port: 60001 Destination port: 36985
589	74.122989	172.16.6.11	172.16.6.15	UDP	118	00000000000000d968a751b70	Source port: 36985 Destination port: 60001
590	77.073827	172.16.6.15	172.16.6.11	UDP	111	800000000000001381ab06df23	Source port: 60001 Destination port: 36985
591	77.126982	172.16.6.11	172.16.6.15	UDP	117	00000000000000dafa41f791d	Source port: 36985 Destination port: 60001
592	80.076797	172.16.6.15	172.16.6.11	UDP	117	80000000000000139c1ae9552b	Source port: 60001 Destination port: 36985
593	83.080343	172.16.6.15	172.16.6.11	UDP	108	8000000000000013a0fc0538b9	Source port: 60001 Destination port: 36985
594	86.083944	172.16.6.15	172.16.6.11	UDP	110	8000000000000013b94abb78e5	Source port: 60001 Destination port: 36985
595	89.087652	172.16.6.15	172.16.6.11	UDP	107	8000000000000013ce2835b7ea	Source port: 60001 Destination port: 36985
596	92.091240	172.16.6.15	172.16.6.11	UDP	112	8000000000000013df2c1c39b0	Source port: 60001 Destination port: 36985
597	95.094905	172.16.6.15	172.16.6.11	UDP	106	8000000000000013e110720ad6	Source port: 60001 Destination port: 36985
598	98.098624	172.16.6.15	172.16.6.11	UDP	115	8000000000000013f474011fbf	Source port: 60001 Destination port: 36985
599	101.102344	172.16.6.15	172.16.6.11	UDP	112	8000000000000014025fa1505c	Source port: 60001 Destination port: 36985
600	104.106053	172.16.6.15	172.16.6.11	UDP	113	800000000000001411a40bc3c1	Source port: 60001 Destination port: 36985
601	107.109623	172.16.6.15	172.16.6.11	UDP	106	80000000000000142154da76a1	Source port: 60001 Destination port: 36985
602	110.115954	172.16.6.15	172.16.6.11	UDP	112	800000000000001431fc91d298	Source port: 60001 Destination port: 36985
603	113.119676	172.16.6.15	172.16.6.11	UDP	110	80000000000000144b8c8a6205	Source port: 60001 Destination port: 36985
604	116.123411	172.16.6.15	172.16.6.11	UDP	123	80000000000000145161b07cef	Source port: 60001 Destination port: 36985
605	119.127137	172.16.6.15	172.16.6.11	UDP	109	80000000000000146aa9001969	Source port: 60001 Destination port: 36985
606	122.157034	172.16.6.11	172.16.6.15	UDP	111	00000000000000e9275967ca1	Source port: 47563 Destination port: 60001
607	122.164498	172.16.6.15	172.16.6.11	UDP	121	800000000000001473c86128e3	Source port: 60001 Destination port: 47563
608	125.159484	172.16.6.11	172.16.6.15	UDP	117	00000000000000eafdf8caa288	Source port: 36220 Destination port: 60001
609	125.164617	172.16.6.15	172.16.6.11	UDP	119	8000000000000014871a4411cf	Source port: 60001 Destination port: 36220
610	128.168239	172.16.6.15	172.16.6.11	UDP	110	80000000000000149e8b57a818	Source port: 60001 Destination port: 36220
611	128.168635	172.16.6.11	172.16.6.15	UDP	120	00000000000000eb713351a3e	Source port: 36220 Destination port: 60001
612	131.160988	172.16.6.11	172.16.6.15	UDP	108	00000000000000ecb211a7ddd	Source port: 36220 Destination port: 60001
613	131.167745	172.16.6.15	172.16.6.11	UDP	120	8000000000000014a1a0f02806	Source port: 60001 Destination port: 36220
614	134.163299	172.16.6.11	172.16.6.15	UDP	122	00000000000000ed1bee26fcb	Source port: 36220 Destination port: 60001
615	134.168256	172.16.6.15	172.16.6.11	UDP	123	8000000000000014bbd95a9b3b	Source port: 60001 Destination port: 36220
616	137.171908	172.16.6.15	172.16.6.11	UDP	111	8000000000000014c993a52750	Source port: 60001 Destination port: 36220



# Vor- und Nachteile?

## Vorteile:

- Mosh hält die Verbindung aufrecht, auch wenn der Benutzer „roamt“ und eine andere IP-Adresse erhält, beispielsweise wenn der Benutzer eines WLAN in ein anderes WLAN oder von WLAN zu UMTS wechselt.
- Mosh hält die Verbindung offen, wenn ein Benutzer die Internetverbindung verliert oder stellt den Client in einen „Schlafmodus“.
- Mosh versucht dem Benutzer sofort zu zeigen, welche Tasten er getippt hat und welche Buchstaben und Wörter er gelöscht hat, um die Latenz auszugleichen. (Speculative Local Echo)



# Vor- und Nachteile?

## Nachteile:

- Mosh stellt Anforderungen an den Server
- UDP bidirektional => FW-Probleme
- Portrange 60000 - 61000
- Kein Scrollback => screen/tmux einsetzen
- kein SSH-Agent Forwarding
- kein X11-Forwarding
- kein SCP Copy
- Probleme bei Dual-Stack IPv4 / IPv6



# Bleiben wir im Kontakt



## Mirko Ludeke

Tel. +49 30 40 50 51-72

[m.ludeke@heinlein-support.de](mailto:m.ludeke@heinlein-support.de)

Heinlein Support GmbH

Schwedter Straße 8/9 | 10119 Berlin

[www.heinlein-support.de](http://www.heinlein-support.de)