Ablauf SSHD

* Ssh\_init 🡪globale Datenstrukturen, ohne Handle
* Ssh\_event\_new() 🡪neuer event context für den Arbeits-Thread (nur dieser eine EventContext wird benötigt), wird im SshDemon gespeichert
* Initialisierung der vier Callbacks
  + server (channel\_open, password),
  + generic
  + bind (incoming connection),
  + channel (data+andere) werdem im SshDemon gespeichert
* Virtual File System aufbauen
* Ssh\_bind\_new() 🡪neuer server bind, wird im SshDemon gespreichert
* Dann Setzen mehrerer bind-Optionen, ohne Handle
* Dann bind-callbacks setzen 🡪incoming\_connection
* Dann ssh\_bind\_listen
* Dann ssh\_bind\_set\_blocking(0)
* Dann ssh\_event\_add\_poll(sshevent, ssh\_bind\_get\_poll(sshbind));
* Dann in der ewigen Schleife int error = ssh\_event\_dopoll(sshevent, 1000);

# Ablauf Incoming Connection als bind-Callback

* Suche freien slot für Context
* Context-Objekt und session-Objekt erzeugen und session in Context setzen
* Bind akzeptieren
* Generic-callbacks und server-callbacks setzen
* Zulässige auth-methoden setzen
* Kex starten
* Session zum session management hinzufügen

# Ablauf channel\_open als server-Callback

* ConnectionCtx raussuchen
* Channel-Objekt erzeugen und im ConnectionCtx aus setzen
* Channel-callbacks setzen