Table des matières

sql-01 3
Machine virtuelle 3
Topologie 3
Configuration 3
Système d'exploitation 3
Adressage IP 3
Paramètres réseau et swap dans sysctl 4
Routage et filtrage avec iptables 4
Paquets installés 6
Stockage 17
Sauvegardes des bases 18
MySQL/MariaDB 18
PostgreSQL 18

sql-01

Machine virtuelle

- Partie matérielle (virtuelle) KVM/QEMU
  - 4 cœurs virtuels
  - 16 Go de mémoire RAM
  - 1 disque virtuel de 500 Go
  - Accès au 1 Gbit/s
- Partie logicielle
  - Système d'exploitation : Debian stable
  - Technologies de virtualisation : KVM, QEMU, libvirt
  - Stockage des données : ZFS
  - 1 IPv4 : 192.168.10.6/32
  - 1 IPv6 : 2a01:4f8:231:aa6::6/64

Topologie

Cette VM héberge tous les services de gestion de bases de données de Liberta.

Ce serveur héberge donc les services :

- MySQL/MariaDB
- PostgreSQL
- Redis

Configuration

Système d'exploitation

- Debian stable (Debian 12 « Bookworm »)

Adressage IP

La route vers 10.10.10.0/24 via l'hyperviseur permet l'accès (par VPN) au réseau d'administration des VM.

```
# cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
```

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp1s0
iface enp1s0 inet static
    address 192.168.10.6/24
    gateway 192.168.10.1
    dns-nameservers 213.133.99.99 213.133.100.100 213.133.98.98
    dns-search liberta.vip
    up ip route add 10.10.10.0/24 via 192.168.10.1 dev enp1s0

iface enp1s0 inet6 static
    address 2a01:4f8:231:aa6::6
    netmask 120
    gateway 2a01:4f8:231:aa6::1
```

Paramètres réseau et swap dans sysctl

Dans `/etc/sysctl.d/99-sysctl.conf` nous avons passé la « swappiness » à 0. La mémoire doit être donc complètement saturée avant de commencer à « swapper » sur le disque dur (pratique d'ailleurs sujet à débats dont nous avons conscience). Nous avons également activé le paramètre d'overcommit à 1 pour éviter que Redis ne se prenne les foudres du « Out of Memory Killer » de Linux:

```
vm.swappiness=0

# Disbale transparent hugepages
vm.nr_hugepages = 0
vm.nr_hugepages_mempolicy = 0
vm.hugepages_treat_as_movable = 0
vm.nr_overcommit_hugepages = 0

# Redis recommendation:
vm.overcommit_memory=1
```

Routage et filtrage avec iptables

Le paquet `iptables-persistent` doit avoir été installé pour conserver les modifications du pare-feu entre chaque redémarrage.

Il est bien sûr extrêmement important de sécuriser SSH : interdire le login root avec mot de passe, utiliser de bons algorithmes de chiffrement, changer le port, n'autoriser qu'une IP distante (ou mieux, ne rien autoriser depuis internet et utiliser un VPN) et mettre en place un faux serveur SSH pour que les attaquants perdent leur temps à essayer de se connecter, sans vous faire perdre le vôtre (et ajouter un Fail2Ban évidemment). La recette reste secrète, désolé ! 😉

Cela dit, pour information une configuration similaire à la suivante est en place :

```
# /etc/ssh/sshd_config.d/liberta.conf

# Common parameters:
Port <un_port>
Port <un_autre_port>
AcceptEnv LANG LC_*
ChallengeResponseAuthentication no
KbdInteractiveAuthentication no
PrintMotd no
PasswordAuthentication no
Subsystem sftp /usr/lib/openssh/sftp-server
UsePAM yes
X11Forwarding no

# Port <un_port> configuration for IPv4/IPv6:
Match
Address="<adresse_ipv4_de_confiance>,127.0.0.0/8,<adresse_ipv6_de_confiance>
,fd00::/8" LocalPort=<un_port>
    AllowUsers root <utilisateurice_de_confiance>

# Port <un_autre_port> configuration for IPv4/IPv6:
Match LocalPort=<un_autre_port>
    AllowUsers <utilisateurice_de_confiance>
```

Pour IPv4, l'hyperviseur s'occupe déjà de tout router et rediriger, nous n'avons besoin d'aucune règle.

En revanche, nous disposons d'une IPv6 routable et exposée sur internet ! Nous devons donc avoir des règles de pare-feu actives. Dans `/etc/iptables-persistent/rules.v6` :

```
*filter
-A INPUT -m conntrack --ctstate RELATED,ESTABLISHED  -m comment --comment
"Accepter le trafic basique : ICMP, boucle locale et connexions établies, en
entrée" -j ACCEPT
-A INPUT -i lo  -m comment --comment "Accepter le trafic basique : ICMP,
boucle locale et connexions établies, en entrée" -j ACCEPT
-A INPUT ! -i lo -d ::1/128  -m comment --comment "Accepter le trafic
basique : ICMP, boucle locale et connexions établies, en entrée" -j REJECT
-A INPUT -s 2a01:4f8:231:aa6::/120 -p tcp -m tcp --syn -m conntrack --
ctstate NEW --dport 22 -m comment --comment "Accepter le SSH depuis les
autres VM" -j ACCEPT
-A INPUT -s 2a01:4f8:231:aa6::/120 -p tcp -m tcp -m multiport --dports 10050
-m comment --comment "Accepter les requêtes Zabbix passives (port 10050)
depuis les VM" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type parameter-problem -m comment --comment "On
accepte l'ICMPv6 indispensable au fonctionnement d'IPv6" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type echo-request -m comment --comment "On
accepte l'ICMPv6 indispensable au fonctionnement d'IPv6" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type echo-reply -m comment --comment "On accepte
l'ICMPv6 indispensable au fonctionnement d'IPv6" -j ACCEPT
```

```
-A INPUT -p icmpv6 --icmpv6-type router-advertisement -m hl --hl-eq 255 -m
comment --comment "On accepte l'ICMPv6 indispensable au fonctionnement
d'IPv6" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type router-solicitation -m hl --hl-eq 255 -m
comment --comment "On accepte l'ICMPv6 indispensable au fonctionnement
d'IPv6" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type neighbour-advertisement -m hl --hl-eq 255 -
m comment --comment "On accepte l'ICMPv6 indispensable au fonctionnement
d'IPv6" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type neighbour-solicitation -m hl --hl-eq 255 -m
comment --comment "On accepte l'ICMPv6 indispensable au fonctionnement
d'IPv6" -j ACCEPT
-A INPUT -p icmpv6 --icmpv6-type echo-request -m conntrack --ctstate NEW -m
limit --limit 1/s --limit-burst 1 -m comment --comment "On refuse les trop
nombreux ping" -j ACCEPT
-A INPUT -s 2a01:4f8:231:aa6::/120 -p tcp -m tcp --syn -m conntrack --
ctstate NEW -m multiport --dports 5432,3306,6379 -m comment --comment
"Accepter les connexions mysql, postgresql et redis depuis les autres VM" -j
ACCEPT
-A INPUT -p icmpv6 --icmpv6-type echo-request -m comment --comment "On
refuse les trop nombreux ping" -j DROP
-A INPUT -m conntrack --ctstate INVALID -m comment --comment "On refuse tout
le reste en entrée" -j DROP
-A INPUT -m comment --comment "On refuse tout le reste en entrée" -j REJECT
COMMIT
```

Paquets installés

Nous utilisons le dépôts pgdg pour diposer des paquets officielsde PostgreSQL, dans `/etc/apt/sources.list.d/pgdg.list` :

```
deb http://apt.postgresql.org/pub/repos/apt bookworm-pgdg main
```

La liste des paquets :

```
# dpkg -l | grep '^i' |awk '{ print $2 }' | sed '/^$/d'| sort
adduser
apparmor
apt
apt-listchanges
apt-utils
aspell
aspell-fr
base-files
base-passwd
bash
bash-completion
bind9-host
bind9-libs:amd64
```

```
bsdextrautils
bsdutils
busybox
bzip2
ca-certificates
console-setup
console-setup-linux
coreutils
cpio
cron
cron-daemon-common
curl
dash
dbus
dbus-bin
dbus-daemon
dbus-session-bus-common
dbus-system-bus-common
dbus-user-session
debconf
debconf-i18n
debian-archive-keyring
debian-faq
debianutils
dictionaries-common
diffutils
dirmngr
discover
discover-data
distro-info-data
dmidecode
dmsetup
doc-debian
dpkg
e2fsprogs
eject
emacsen-common
ethtool
exuberant-ctags
fail2ban
fdisk
file
findutils
firmware-linux-free
fontconfig-config
fonts-dejavu-core
galera-4
gawk
gcc-10-base:amd64
gcc-12-base:amd64
gcc-8-base:amd64
```

```
gdal-data
gdal-plugins
gdbm-l10n
gettext-base
gnupg
gnupg-l10n
gnupg-utils
gpg
gpg-agent
gpgconf
gpgsm
gpgv
gpg-wks-client
gpg-wks-server
grep
groff-base
grub2-common
grub-common
grub-pc
grub-pc-bin
guile-3.0-libs:amd64
gzip
hdparm
hostname
htop
ifrench-gut
iftop
ifupdown
inetutils-telnet
init
initramfs-tools
initramfs-tools-core
init-system-helpers
installation-report
iotop
iproute2
iptables
iptables-persistent
iputils-ping
isc-dhcp-client
isc-dhcp-common
iso-codes
ispell
kbd
keyboard-configuration
keyutils
klibc-utils
kmod
krb5-locales
laptop-detect
less
```

```
libacl1:amd64
libaec0:amd64
libaom3:amd64
libapparmor1:amd64
libapt-inst2.0:amd64
libapt-pkg5.0:amd64
libapt-pkg6.0:amd64
libargon2-1:amd64
libarmadillo11
libarpack2:amd64
libaspell15:amd64
libassuan0:amd64
libatomic1:amd64
libattr1:amd64
libaudit1:amd64
libaudit-common
libblas3:amd64
libblkid1:amd64
libblosc1:amd64
libbpf1:amd64
libbrotli1:amd64
libbsd0:amd64
libbz2-1.0:amd64
libc6:amd64
libcap2:amd64
libcap2-bin
libcap-ng0:amd64
libc-bin
libcbor0.8:amd64
libcfitsio10:amd64
libcgi-fast-perl
libcgi-pm-perl
libc-l10n
libclone-perl:amd64
libcom-err2:amd64
libcommon-sense-perl:amd64
libconfig-inifiles-perl
libcrypt1:amd64
libcryptsetup12:amd64
libcurl3-gnutls:amd64
libcurl4:amd64
libdav1d6:amd64
libdaxctl1:amd64
libdb5.3:amd64
libdbd-mysql-perl:amd64
libdbd-pg-perl
libdbi-perl:amd64
libdbus-1-3:amd64
libde265-0:amd64
libdebconfclient0:amd64
libdeflate0:amd64
```

```
libdevmapper1.02.1:amd64
libdiscover2
libdns-export1104
libedit2:amd64
libefiboot1:amd64
libefivar1:amd64
libelf1:amd64
libencode-locale-perl
libestr0:amd64
libevent-core-2.1-7:amd64
libexpat1:amd64
libext2fs2:amd64
libfastjson4:amd64
libfcgi0ldbl:amd64
libfcgi-bin
libfcgi-perl
libfdisk1:amd64
libffi6:amd64
libffi8:amd64
libfido2-1:amd64
libfile-find-rule-perl
libfontconfig1:amd64
libfreetype6:amd64
libfreexl1:amd64
libfstrm0:amd64
libfuse2:amd64
libfyba0:amd64
libgc1:amd64
libgcc1:amd64
libgcc-s1:amd64
libgcrypt20:amd64
libgdal32
libgdbm6:amd64
libgdbm-compat4:amd64
libgeos3.11.1:amd64
libgeos-c1v5:amd64
libgeotiff5:amd64
libgfortran5:amd64
libgif7:amd64
libglib2.0-0:amd64
libglib2.0-data
libgmp10:amd64
libgnutls30:amd64
libgpg-error0:amd64
libgpm2:amd64
libgssapi-krb5-2:amd64
libhdf4-0-alt
libhdf5-103-1:amd64
libhdf5-hl-100:amd64
libheif1:amd64
libhogweed4:amd64
```

```
libhogweed6:amd64
libhtml-parser-perl:amd64
libhtml-tagset-perl
libhtml-template-perl
libhttp-date-perl
libhttp-message-perl
libicu72:amd64
libidn11:amd64
libidn2-0:amd64
libio-html-perl
libio-pty-perl
libip4tc0:amd64
libip4tc2:amd64
libip6tc0:amd64
libip6tc2:amd64
libipc-run-perl
libiptc0:amd64
libisc-export1100:amd64
libjbig0:amd64
libjemalloc2:amd64
libjpeg62-turbo:amd64
libjson-c3:amd64
libjson-c5:amd64
libjson-perl
libjson-xs-perl
libk5crypto3:amd64
libkeyutils1:amd64
libklibc:amd64
libkmlbase1:amd64
libkmldom1:amd64
libkmlengine1:amd64
libkmod2:amd64
libkrb5-3:amd64
libkrb5support0:amd64
libksba8:amd64
liblapack3:amd64
liblcms2-2:amd64
libldap-2.5-0:amd64
libldap-common
liblerc4:amd64
libllvm16:amd64
libllvm19:amd64
liblmdb0:amd64
liblocale-gettext-perl
liblockfile-bin
liblognorm5:amd64
libltdl7:amd64
liblwp-mediatypes-perl
liblz4-1:amd64
liblzf1:amd64
liblzma5:amd64
```

```
liblzo2-2:amd64
libmagic1:amd64
libmagic-mgc
libmariadb3:amd64
libmaxminddb0:amd64
libmd0:amd64
libminizip1:amd64
libmnl0:amd64
libmount1:amd64
libmpfr6:amd64
libncurses6:amd64
libncursesw6:amd64
libndctl6:amd64
libnetcdf19:amd64
libnetfilter-conntrack3:amd64
libnettle6:amd64
libnettle8:amd64
libnewt0.52:amd64
libnfnetlink0:amd64
libnfsidmap1:amd64
libnftnl11:amd64
libnghttp2-14:amd64
libnl-3-200:amd64
libnl-genl-3-200:amd64
libnpth0:amd64
libnsl2:amd64
libnspr4:amd64
libnss3:amd64
libnss-nis:amd64
libnss-nisplus:amd64
libnss-systemd:amd64
libnuma1:amd64
libnumber-compare-perl
libodbc2:amd64
libodbcinst2:amd64
libogdi4.1
libopenjp2-7:amd64
libp11-kit0:amd64
libpam0g:amd64
libpam-modules:amd64
libpam-modules-bin
libpam-runtime
libpam-systemd:amd64
libpcap0.8:amd64
libpci3:amd64
libpcre2-8-0:amd64
libpcre3:amd64
libperl5.36:amd64
libpipeline1:amd64
libpmem1:amd64
libpng16-16:amd64
```

```
libpoppler126:amd64
libpopt0:amd64
libpq5:amd64
libproc2-0:amd64
libprocps7:amd64
libproj25:amd64
libprotobuf-c1:amd64
libpsl5:amd64
libpython3.11-minimal:amd64
libpython3.11-stdlib:amd64
libpython3-stdlib:amd64
libqhull-r8.0:amd64
libquadmath0:amd64
libreadline8:amd64
libregexp-ipv6-perl
librtmp1:amd64
librttopo1:amd64
libsasl2-2:amd64
libsasl2-modules:amd64
libsasl2-modules-db:amd64
libseccomp2:amd64
libselinux1:amd64
libsemanage2:amd64
libsemanage-common
libsensors5:amd64
libsensors-config
libsepol1:amd64
libsepol2:amd64
libsigsegv2:amd64
libslang2:amd64
libsmartcols1:amd64
libsnappy1v5:amd64
libsodium23:amd64
libspatialite7:amd64
libsqlite3-0:amd64
libss2:amd64
libssh2-1:amd64
libssl1.1:amd64
libssl3:amd64
libstdc++6:amd64
libsuperlu5:amd64
libsystemd0:amd64
libsystemd-shared:amd64
libsz2:amd64
libtasn1-6:amd64
libterm-readkey-perl
libtext-charwidth-perl:amd64
libtext-glob-perl
libtext-iconv-perl:amd64
libtext-template-perl
libtext-wrapi18n-perl
```

```
libtiff6:amd64
libtimedate-perl
libtinfo6:amd64
libtirpc3:amd64
libtirpc-common
libtypes-serialiser-perl
libuchardet0:amd64
libudev1:amd64
libunistring2:amd64
liburing2:amd64
liburiparser1:amd64
liburi-perl
libusb-1.0-0:amd64
libuuid1:amd64
libuv1:amd64
libwebp7:amd64
libwrap0:amd64
libx11-6:amd64
libx11-data
libx265-199:amd64
libxau6:amd64
libxcb1:amd64
libxdmcp6:amd64
libxerces-c3.2:amd64
libxext6:amd64
libxml2:amd64
libxmuu1:amd64
libxslt1.1:amd64
libxtables12:amd64
libxxhash0:amd64
libz3-4:amd64
libzstd1:amd64
linux-base
linux-image-6.1.0-39-amd64
linux-image-6.1.0-40-amd64
linux-image-amd64
locales
login
logrotate
logsave
lsb-base
lsb-release
lsof
mailcap
make-guile
man-db
manpages
manpages-fr
mariadb-backup
mariadb-client
mariadb-client-core
```

```
mariadb-common
mariadb-plugin-provider-bzip2
mariadb-plugin-provider-lz4
mariadb-plugin-provider-lzma
mariadb-plugin-provider-lzo
mariadb-plugin-provider-snappy
mariadb-server
mariadb-server-core
mawk
media-types
mime-support
mount
mysql-common
nano
ncal
ncurses-base
ncurses-bin
ncurses-term
netbase
netcat-traditional
netfilter-persistent
net-tools
nfs-common
openssh-client
openssh-server
openssh-sftp-server
openssl
os-prober
passwd
pci.ids
pciutils
perl
perl-base
perl-modules-5.36
pg-activity
pinentry-curses
poppler-data
postgis
postgis-doc
postgresql
postgresql-17
postgresql-17-rum
postgresql-18
postgresql-18-jit
postgresql-client-17
postgresql-client-18
postgresql-client-common
postgresql-common
postgresql-common-dev
postgresql-contrib
powermgmt-base
```

```
procps
proj-bin
proj-data
psmisc
publicsuffix
pv
python3
python3.11
python3.11-minimal
python3-apt
python3-attr
python3-blessed
python3-certifi
python3-cffi-backend:amd64
python3-chardet
python3-charset-normalizer
python3-cryptography
python3-debconf
python3-debian
python3-debianbts
python3-httplib2
python3-humanize
python3-idna
python3-minimal
python3-pkg-resources
python3-psutil
python3-psycopg2
python3-pycurl
python3-pyinotify
python3-pymysql
python3-pyparsing
python3-pysimplesoap
python3-reportbug
python3-requests
python3-six
python3-systemd
python3-urllib3
python3-wcwidth
python-apt-common
python-is-python3
qemu-guest-agent
readline-common
redis
redis-server
redis-tools
reportbug
rpcbind
rsync
rsyslog
runit-helper
sed
```

```
sensible-utils
shared-mime-info
socat
ssl-cert
sysstat
systemd
systemd-sysv
systemd-timesyncd
sysvinit-utils
tar
task-french
tasksel
tasksel-data
task-ssh-server
tcpdump
traceroute
tzdata
ucf
udev
unixodbc-common
usbutils
usrmerge
util-linux
util-linux-extra
util-linux-locales
vim
vim-common
vim-runtime
vim-tiny
wamerican
wfrench
wget
whiptail
whois
xauth
xdg-user-dirs
xkb-data
xxd
xz-utils
zabbix-agent2
zlib1g:amd64
zstd
```

Stockage

Nous montons les partages ZFS de nos backups MySQL et PostgreSQL :

Dans `/etc/fstab` :

```
# mysql_backups ZFS NFS share:
```

```
192.168.10.1:/zdata/mysql_data  /mysql_backups  nfs  auto  0 0
# postgresql_backups ZFS NFS share:
192.168.10.1:/zdata/postgresql_data /postgresql_backups  nfs  auto  0 0
```

Sauvegardes des bases

La plus critique et souvent la plus négligée des tâches est souvent cette partie, étrangement (Liberta en fait aussi partie et a déjà eu quelques sueurs froides par le passé).

Nous utiliserons les outils standards fournis par MariaDB et PostgreSQL, bien plus fiables et adaptés que si nous avions compté sur nos partages ZFS (entre les contraintes de NFS et la multitude de paramètres de ZFS pour s'adapter au stockage de bases de données, nous avons choisi le plus simple et le plus fiable).

Ici les prénoms ont été modifiés pour préserver leur confidentialité :)

MySQL/MariaDB

Nous utilisons le paquet `mariadb-backup`, lequel permet de faire nos sauvegardes complètes des bases sans avoir à suspendre ou verrouiller nos bases.

Nous avons créé dans MySQL un « super admin » qui peut accéder à toutes les bases :

```
mysql
MariaDB [(none)]> CREATE USER 'libertasuperuser'@'%' IDENTIFIED BY
'motdepassecomplexe';
MariaDB [(none)]> grant all privileges on *.* to 'libertasuperuser'@'%' with
grant option;
MariaDB [(none)]> flush privileges;
```

Notre tâche planifiée qu'on créé dans une tâche `cron`. Nous gardons les 7 derniers jours localement pour en disposer facilement en cas de coup dur, ensuite ce sont nos snapshots ZFS qui s'occupent du reste :

```
@daily find /mysql_backups/ -type d -ctime +7 -exec rm -rf {}+ && mariadb-
backup --backup --target-dir=/mysql_backups/$(date +"%Y%m%d-%H%M%S") --
user='libertasuperuser' --password='motdepassecomplexe'
```

PostgreSQL

Pour PostgreSQL, il faut également créer un « super-admin » avec tous les droits et corriger les droits et permissions. Notre serveur écoute sur son IPv4 192.168.10.6 et notre super admin doit disposer des droits de réplication pour que `pg_basebackup` fonctionne :

```
su - postgres
postgres@sql-01:~$ psql
postgres=# CREATE USER libertasuperuser WITH SUPERUSER;
```

```
postgres=# ALTER USER libertasuperuser WITH PASSWORD 'motdepassecomplexe';
postgres=# ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT ALL ON TABLES TO
libertasuperuser;
postgres=# ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT ALL ON SEQUENCES
TO libertasuperuser;
postgres=# ALTER ROLE libertasuperuser WITH REPLICATION;
postgres=# SELECT rolreplication FROM pg_roles WHERE rolname =
'libertasuperuser';
postgres=# \q
```

Dans `/etc/postgresql/<version>/main/pg_hba.conf`, nous définissons les permissions :

```
# "local" is for Unix domain socket connections only
local   all             all                                     peer
# IPv4 local connections:
host    all             all             127.0.0.1/32            scram-
sha-256
# IPv6 local connections:
host    all             all             ::1/128                 scram-
sha-256
# On permet à Peertube de se connecter :
host    all             all             192.168.10.5/32           md5
# On permet à Funkwhale de se connecter :
host    all             all             192.168.10.9/32           md5
# On permet à Zabbix/OpenObserve de se connecter :
host    all             all             192.168.10.250/32         md5
# On permet au superadmin de se connecter localement :
host    all             libertasuperuser  127.0.0.1/32            md5

# Allow replication connections from localhost, by a user with the
# replication privilege.
local   replication     all                                     peer
host    replication     all             127.0.0.1/32            scram-
sha-256
host    replication     all             ::1/128                 scram-
sha-256
host    replication     libertasuperuser     192.168.10.6/32     md5
```

Nous utilisons l'outil fourni en standard `pg_basebackup` dans une tâche planifiée `cron` pour l'utilisateur standard `postgres`. Idem que pour MySQL, nous conservons 7 jours en local et les snapshots ZFS s'occupent du reste :

```
export PGPASSWORD='motdepassecomplexe' && pg_basebackup -h 192.168.10.6 --
username=libertasuperuser --no-password -D /postgresql_backups/$(date
+"%Y%m%d-%H%M%S")
```

From:
https://doc.liberta.vip/ - **Documentation Liberta**

Permanent link:
https://doc.liberta.vip/tech/sql-01?rev=1761233562

Last update: **23/10/2025 17:32**