

Sample Scripts

????

- [ip.sh](#) - ?? Public IP ????
- [whoamifuck.sh](#) - Linux ???????
- [dump_stats.sh](#) - ?? Linux/Mac ????

Auto-rar.sh

?????????? (*.part0XX.rar)

```
#!/bin/bash

echo "-> Started: "`date +%m/%d/%y\ %H:%M\ %Z`
echo "-> As:"`whoami`

i=1
ARCHIVES="$@"
for f in $ARCHIVES; do
    PARTCHECK=$(expr "$f" : ".*\([Pp][Aa][Rr][Tt][0-9]\+\.\[Rr][Aa][Rr]\)")
    RARCHECK=$(expr "$f" : "\(.*\.\[Rr][Aa][Rr]\)")
    echo "-> Processing($i of $#): $f"

    if [ "$PARTCHECK" ] && [ -f $f ]; then
        echo "-> Extracting Multipart Archive"
        unrar x $f
        if [ $? -eq 0 ]; then
            FILES=$(expr "$f" : "\(.*\[Pp][Aa][Rr][Tt]\).*.")
            echo "-> Extraction Successful"
            echo "-> Removing $FILES*.rar"
            rm $FILES*.rar
        else
            echo "-> **Extraction Failed"
            exit 1
        fi
    fi
done
```

```

else
  if [ "$RARCHECK" ] && [ -f $f ]; then
    echo "-> Extracting Single Archive"
    unrar x $f
    if [ $? -eq 0 ]; then
      echo "-> Extraction Successful"
      echo "-> Removing $f"
      rm $f
    else
      echo "-> **Extraction Failed"
      exit 1
    fi
  fi
fi
echo ""
i=$(( i+1 ))
done

```

Domaincheck.sh

```

#!/bin/bash

#Specify all the domains you want to check
DOMAINS="google.com dribbble.com facebook.com youtube.com"

current_epoch=`date +%s`
for dm in $DOMAINS
do
  expiry_date=`whois $dm | egrep -i "Expiration Date:|Expires on"| head -1 | awk '{print $NF}'`
  echo -n " $dm - Expires on $expiry_date "
  expiry_epoch=`date --date="$expiry_date" +%s`
  epoch_diff=`expr $expiry_epoch - $current_epoch`
  days=`expr $epoch_diff / 86400`
  echo " $days days remaining. "
done

```

Check my.cnf and file permissions

```
# Check for .my.cnf
if [ ! -f ~/.my.cnf ]; then
    echo "It seems MySQL credentials are missing."
    echo "Please create a ~/.my.cnf file with the following structure:"
    echo ""
    echo "[client]"
    echo "user=your_username"
    echo "password=your_password"
    echo ""
    echo "Ensure the file permissions are secure: chmod 600 ~/.my.cnf"
    exit 1
else
    # Check file permissions
    file_perm=$(stat -c "%a" ~/.my.cnf)
    if [ "$file_perm" -ne 600 ]; then
        echo "Error: ~/.my.cnf permissions are $file_perm. Please set them to 600 using 'chmod 600
~/.my.cnf'."
        exit 1
    fi
    echo "Credentials found in ~/.my.cnf, but the command still failed."
    echo "MySQL admin output:"
    echo "$output"
    echo "Please verify your credentials or server status."
    exit 1
fi
```

Revision #7

Created 2020-06-01 07:44:00 CST by A-Lang (Admin)

Updated 2025-04-02 15:44:20 CST by A-Lang (Admin)