

# While loops

## Run for period of time

```
#!/bin/bash

runtime="5 minute"
endtime=$(date -ud "$runtime" +%s)

while [[ $(date -u +%s) -le $endtime ]]
do
    echo "Time Now: `date +%H:%M:%S`"
    echo "Sleeping for 10 seconds"
    sleep 10
done
```

**Resource:** <https://www.golinuxcloud.com/run-while-loop-until-specific-time-shell-bash/>

## Infinite loop

```
item=true
while [ $item = true ]; do echo 'bla'; done
```

## While value is an empty string

```
value=""
while [[ -z "$value" ]]; do
    echo "value is empty"
    # here's where we break out
    if [[ "$value" ]]; then
        echo "exiting loop because value has been set to $value"
        break
    fi
done
```

# Read each line of a file

```
while read p; do
  echo "$p"
done <file.txt
```

**Resource:** <https://stackoverflow.com/questions/1521462/looping-through-the-content-of-a-file-in-bash>

## One-liner equivalent

```
while read t; do echo "$t"; done <bbh_targets.txt
```

# Execute command once per line of piped input

```
command | while read -r line; do command "$line"; done
```

For example, if you want to run a program and send the output to different files using regular expressions:

```
recon asn -t recon_targets.txt | \
while read -r line; do \
  ([[ $line =~ ^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\[/0-9]{2}$ ]] \
  && echo $line | anew ip_ranges.txt) || ([[ $line =~ ^[0-9]{5}$ ]] && \
  echo $line | anew asns.txt); done
```

**Resources:** <https://unix.stackexchange.com/questions/7558/execute-a-command-once-per-line-of-piped-input> [Tool run for the example](#)

---

Revision #2

Created 2023-11-10 06:10:32 UTC by Ryan

Updated 2025-02-12 01:10:55 UTC by Ryan