

Módulo C-Backup_5_min

OPCIÓN A: Verificado en Entornos de Euroskills'25 (cuidado sin timestamp!!!)

Pros: trivial to set up, auto-rotation

Cons: filenames are numbered (not timestamped)

```
conf t
  archive
    path flash:cfg-archive
    write-memory
    time-period 60      ! every 60 minutes
    maximum 5           ! keep last 5
end
```

This will create `flash:cfg-archive-1`, `-2`, ... and prune older ones automatically.

If you **must** have date/time in the filename, use Option B.

OPCIÓN B: No verificado en Entornos de Euroskills'25 (con Timestamp)

Option B: EEM Tcl policy with timestamped filenames + rotation

Pros: filename includes date & hour; keeps last 5 automatically

Cons: slightly more setup

1) (One-time) Tell EEM where your user policies live

```
conf t
  event manager directory user policy flash:
end
```

2) Create the policy file `flash:hourly_backup.tcl`

Paste the following exact content into `flash:hourly_backup.tcl` (via your preferred method: USB, SCP, or `tclsh + puts`).

```
# flash:hourly_backup.tcl
# Hourly running-config backup to flash:configs with timestamped filename.
# Keeps only the latest 5 files.

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

# Run every hour at minute 0
::cisco::eem::event_register_timer cron name hourly_cfg_backup cron_entry "0 * *
* *"
```

```

proc do_backup {} {
    # Build timestamp: YYYYMMDD-HHMM
    set ts [clock format [clock seconds] -format {%Y%m%d-%H%M}]
    set dir "flash:configs"
    set filename "$dir/running-$ts.cfg"

    # Ensure directory exists
    catch {file mkdir $dir}

    # Open CLI
    array set cli [cli_open]
    set fd $cli(fd)
    cli_exec $fd "enable"

    # Save running-config to timestamped file
    # Use /overwrite to avoid prompts if a colliding name ever occurs
    cli_exec $fd "copy /y running-config $filename"

    # Rotation: keep only the newest 5 matching files
    set files [lsort [glob -nocomplain "$dir/running-*.cfg"]]
    set count [llength $files]
    if {$count > 5} {
        # Delete the oldest files, keep last 5
        set to_delete [lrange $files 0 [expr {$count - 6}]]
        foreach f $to_delete {
            catch {file delete $f}
        }
    }

    cli_close $fd
}

# Run
do_backup

```

3) Register the policy so it runs hourly

```

conf t
    event manager policy hourly_backup.tcl type user
end

```