



*Full credit is given to the above companies including the OS that this PDF file was generated!*

## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'team2bond.1' command***

***\$ man team2bond.1***

TEAM2BOND(1) Team to Bonding conversion tool TEAM2BOND(1)

### NAME

team2bond ? Converts a team configuration to a NetworkManager bond configuration

### SYNOPSIS

team2bond [options]

### DESCRIPTION

team2bond is a tool to convert team options to bonding options. During the conversion, team2bond tries to create a bond configuration that has the same or similar functionality as the team.

This tool converts a team JSON format configuration file to NetworkManager nmcli commands that you can use to set up a network bonding device. After verifying the commands and deleting the corresponding team device, run the nmcli commands manually or with parameter --exec to create the NetworkManager connection profile.

Important: Note that deleting the team device makes this connection unavailable until you recreated the bond.

### OPTIONS

--config <team json config file>

Specifies the team JSON format configuration file to convert.

Use the 'teamdctl <TEAMNAME> config dump [actual]' command to create the JSON file.

--rename <interface>

This is a convenient option to replace the original interface name with the specified name. For example, use `--rename bond0` to change the interface name to `bond0`. Note, firewall rules, alias interfaces, and so on, that are tied to the original interface name can break after the renaming an interface because the tool only changes the NetworkManager profile.

`--help` Print a help text to console and exit.

## DANGEROUS OPTION

`--exec` Executes the generated nmcli commands and adds the connections directly instead of printing them to the screen. This parameter is NOT recommended. Double-check the generated commands before you apply them.

## EXAMPLE STEPS

1. Save the current 'team0' ip addresses, firewall rules, alias, etc.
2. Dump the current 'team0' configuration to a JSON file. Using the 'actual' option, team2bond dumps the actual config instead of the initial config:

```
# teamdctl team0 config dump actual > team0.json
```

3. Run the team2bond utility in dry-run mode to display nmcli commands that set up a network bond with similar settings as the team device:

```
# team2bond --config team0.json
```

To convert the current 'team0' configuration to bonding connection profile file and rename the interface name to 'bond0':

```
# team2bond --config team0.json --rename bond0
```

4. Examine these generated nmcli commands, and make sure all parameters are correct.
5. Delete the team0 device by using the teamd or nmcli utility to avoid a device conflict. Then execute the commands the team2bond utility displayed in the previous step manually or use the `--exec` option:

```
# team2bond --config team0.json --rename bond0 --exec
```

## CAVEATS

The tool will not convert any other configuration which might be tied to the current setup. For instance, IP addresses, firewall rules, alias

interfaces, bridges, and so on.

AUTHOR

Hangbin Liu <haliu@redhat.com>

SEE ALSO

nmcli(1), teamdctl(8), teamd.conf(5).

libteam

2021-07-20

TEAM2BOND(1)