NAME

dynmn_tool – control and monitor **Mobile Node**

SYNOPSIS

dynmn_tool [**–p** agent_path] [command]

DESCRIPTION

dynmn_tool provides a simple interface for controlling and monitoring the **Mobile Node.** The tool can be started in the interactive mode if no command is given, or a specific command can be executed on the command line. In the interactive mode the tool prompts user for commands. Commands are specified in section **COMMANDS.**

OPTIONS

–p *agent_path*

Specify agent API interface socket path to use; the default is **/var/run/dynamics_mn_admin**.

command

Execute specified command and exit. If command has parameters, the command should be enclosed in quotation marks. The command is executed only once even if it is a monitoring command and monitoring interval is specified.

COMMANDS

confirm

Confirm that the tunnel is still up be sending a registration request to the Home Agent.

careof Get the current care-of address for the mobile. This is the address of the Foreign Agent, we are connected to.

tunnel Alternatively, command **connect** can be used for the same purpose.

The plain command **tunnel** initiates the registration to a Foreign Agent if the Mobile Node is currently in the **disconnected** state. The tunneling mode to be requested is specified in the **dynmnd.conf** file, see **dynmnd.conf**(5).

tunnel *HA*

This command makes the Mobile Node attempt direct tunneling connection to the Home Agent with full tunneling. The Home Agent IP address specified in the **dynmnd.conf** will be used. The Mobile Node must be operating with a co-located care-of address (i.e. a publicly routable address from the foreign network).

disconnect

Disconnect from tunneling system and move the Mobile Note to the **disconnected** state (i.e. not trying to find new Foreign Agents).

status Retrieve status information from the Mobile Node.

status *interval*

If interval is specified, command is run in monitoring mode and interval specifies time between successive calls in seconds. To stop monitoring press CTRL-C.

update *ip_addr*
Perform location update and use *ip_addr* as our own address when communicating with local network. If *ip_addr* is same as our home address, tunnel is dropped and mobile is connected directly to network using it's own IP address.

update *interface*
Perform location update and use the primary IP address of the given *interface*

update Perform location update and use the primary IP address of the default interface (config option).

quit Quit dynmn_tool program.

help *[command]*
Show help for a specific command. If command is not specified, all available commands will be listed.

force *fa_addr*
Force the Mobile Node to connect to a Foreign Agent with specific IP address. All other Foreign Agents will be rejected. If *fa_addr* is set to 0.0.0.0, no forcing is applied and any of the available **Foreign Agents** can be used.

list Lists the recently heard Foreign Agents.

list *interval*
If interval is specified, command is run in monitoring mode and interval specifies time between successive calls in seconds. To stop monitoring press CTRL-C.

path Change the API socket path.

show *fa-addr*
Shows detailed information about the heard Foreign Agent.

wireless *channel* **[interface channel]**
API interface for wireless extensions. Query or change the channel (frequency) of the wireless interface card. *interface* is the wireless interface name. *channel* is the channel number (1-14).

policy *[policy on|off]*
dynmn_tool can be used to modify handoff policies. Without arguments *policy* will list currently available policies and their status (on or off). You can change the status for policy by giving second and third argument respectively.

Short-interval

When ON monitor uses specific aging for agent advertisements. If agent advertisement is older than one third of the lifetime plus one second the entry is marked old and is not used as first choice when new FA is selected.

Newest-FA

This mode can be used in infrastructure mode. When ON the FA with newest agent advertisement is used. Newest one is the entry that has been added lastly.

Eager-switching

When ON this policy disables averages and thresholds. It's meant for eager switching between FAs. It enables MN to react faster to link quality changes, but it's not useful at usual situations as the link quality values may change even with no movement.

monitor *[variable [value]]*

dynmn_tool can be used to modify dynamically mobile node's monitor module. Without arguments *monitor* will list currently available and modifiable variables. The second argument will specify the variable. If third argument is not given, dynmn_tool will display the value for that variable. With third argument the value can be changed.

Threshold *percent*

If current FA has at least *percent* link quality compared to the maximum then balance priorities. If threshold is 0 monitor will not change FA in the wireless side. If threshold is 100 monitor will use the best available FA in wireless side. The range is 0 - 100 percent.

Default-channel *channel*

When wireless interfaces are detected the default channel is made *channel.* Range is 1 - 14.

Old-FA-factor *percent*

If *Short-interval* policy is ON the *Old-FA-factor* is used to decrease the priority of FAs that have old agent advertisements. Range 0 - 100 percent.

Min-balance *percent*

If the link quality is below *Min-balance* percent of the maximum, monitor doesn't balance priorities. Range 0 - 100 percent.

Worst-min-time *seconds*

Wireless subsystem of the Linux kernel supports 8 simultaneous iwspy (see iwspy(8)) entries. If more than 8 FAs (or Access Points) are present monitor will change the iwspy entries in kernel in certain intervals. The entry with worst link quality is dropped, but every entry will be monitored at least *Worst-min-time* seconds.

Worst-max-time *seconds*

Wireless subsystem of the Linux kernel supports 8 simultaneous iwspy (see iwspy(8)) entries. If all 8 entries are used and monitor wants new iwspy entry the worst is usually dropped. But if there exists an entry that has zero link quality and has been monitored at least *Worst-max-time* seconds that entry is dropped from kernel.

register *device_info_socket_path*

This API call is not meant to be called from dynmn_tool, but is an interface for device_info(8) daemon. It registers the socket path that can be used to connect device_info(8) daemon.

BUGS

Timeout values for API commands are set to infinity. To interrupt a too long lasting command, use CTRL−C.

AUTHOR AND COPYRIGHT

Dynamics group <dynamics@cs.hut.fi>

SEE ALSO

dynmnd(8), **dynmnd.conf**(5), **dynfad**(8), **dynhad**(8), **dynamics**(8)