

IRC/Discord Commands Tutorial

4 posts • Page 1 of 1

03 Feb 2017, 19:55 **1 66**

#Board index > AutoHotkey > Scripts and Functions

[LogonDesktop] AHK on the lock/logon screen: control Spotify with the media keys there, e.g.

Post Reply 5 Search this topic... Q \$

qwerty12

Posts: 468

Joined: 04 Mar 2016

GitHub: qwerty12

Contact: •••

Warnings abound:

- · The little I know about WIndows security I learnt while writing this. There's a good possibility that running this weakens the security of your system!
- I've only tested this with on my Windows 10 laptop and a Windows 8 virtual machine. (Both had UAC on, I'm not sure what effect turning it off has on this.) Running AHKLogonMediaKeys.ahk elevated/as admin will make the script start itself as SYSTEM then and there and also on startup. Test in a virtual machine.
- · Windows XP + Vista won't work with this. (Fixing it for Vista shouldn't be hard, I think it's just the use of WTSEnumerateSessionsExW in AHKLogonMediaKeys that needs changing.) Under XP, you could try this
- Unicode builds of AutoHotkey only. 32-bit builds should be fine, but I mostly tested this with the 64-bit version

 AHKL ogopMediaKeys abk starts the process that rups in the user session with uiAccess privileges. Comment out the relevant line in AHKL ogopMediaKeys to Jaunch with normal integrity.
- AHKLogonMediaKeys.ahk starts the process that runs in the user session with uiAccess privileges. Comment out the relevant line in AHKLogonMediaKeys to launch with normal integrity AHKLogonMediaKeys.ahk isn't as clean as I'd like it to be
- This won't start on Remote Desktop sessions (which is a plus in my case)

I didn't try with a compiled script

LogonDesktop.ahk contains helper functions for creating a Task Scheduler task that runs as SYSTEM, launching a script on a different Desktop etc. It is used by AHKLogonMediaKeys.ahk, which enables your keyboard's media keys to work with Spotify when the computer is locked. (And since I use the console logon screen of Windows 10, it also enables the volume keys to start working thereagain.)

AHKLogonMediaKeys.ahk works like this: it runs in the Services session to watch for new sessions (each instance of the logon screen gets its own) and will then start an instance of itself running on that session's logon screen. That instance will then, in turn, start another instance of itself running as the person logged on in that session that listens to messages sent to the named pipe itcreates. The instance running on the the logon/lock screen will typically be the one writing to the named pipe.

If the AHK process it's in is running as SYSTEM (which it will be when the Task Scheduler starts it), it determines what session it's running under:

If it's on the services session (0), it tracks new sessions so that it can start another instance of itself on the Winlogon desktop of that session by: enumerating the sessions already in existence

listening to WM_WTSSESSION_CHANGE to get word of new sessions

Otherwise, in any other session (which will be the case for the instances started on the Winlogon desktops), it attempts to:

another instance of itself under your user account that listens for messages sent to the named pipe it creates

if that fails (Winlogon desktop exists, but nobody's logged on there), it listens for WM_WTSSESSION_CHANGE messages particular to that session, waiting for a logon, and then starts the listener under your accountregisters media hotkeys on the Winlogon desktop - the handler will attempt to write the listener's pipe with the name of the key that was pressed

Otherwise, it assumes it's running under a user session. There it works like this:

if on the command line it has a name to give to the named pipe it will create (which it will when started by the Winlogon instance), it creates a pipe and listens for messages but if not:

if the script is elevated, it will add a task to the Task Scheduler that's triggered to run at startup and then attempt to run it so that the script gets started as SYSTEM under session 0if it isn't, it asks for elevation and we're at ^

