Simulating keypress events on Android

>> adb , Android , CodeProject , keyevents , Programming

Modern day smart-phones have already begun to migrate from the traditional "a-button-for-every-need" approach to the "huge-display-cum-touchscreen" form-factors. Android phones are no exception. But, traditional buttons are still reqd. for a few oft used functions (power,back,home,menu etc.) And smart-phones continue to have them alongwith the primary touch-based-UI.

Android-OS provides a very easy method to simulate key/button press/release events via software. You might ask why do we need a software to generate the events when a hardware button is already present on the device. Here's why:

- During development/testing of the button-drivers itself.
- To implement automated rigorous tests. (**MonkeyTest**?)
- To implement/interface additional custom software keyboards.
- Just because we can!

The reason why i am doing it today is **REASON NUMBER 4**.

Now, the basic goal of this exercise is extremely simple:

> Q. How to generate a hardware-button-press event
> WITHOUT actually pressing any key on the device?

Let us first understand what happens when a hardware button is pressed. Here is what happens when you press a button on an Android device:

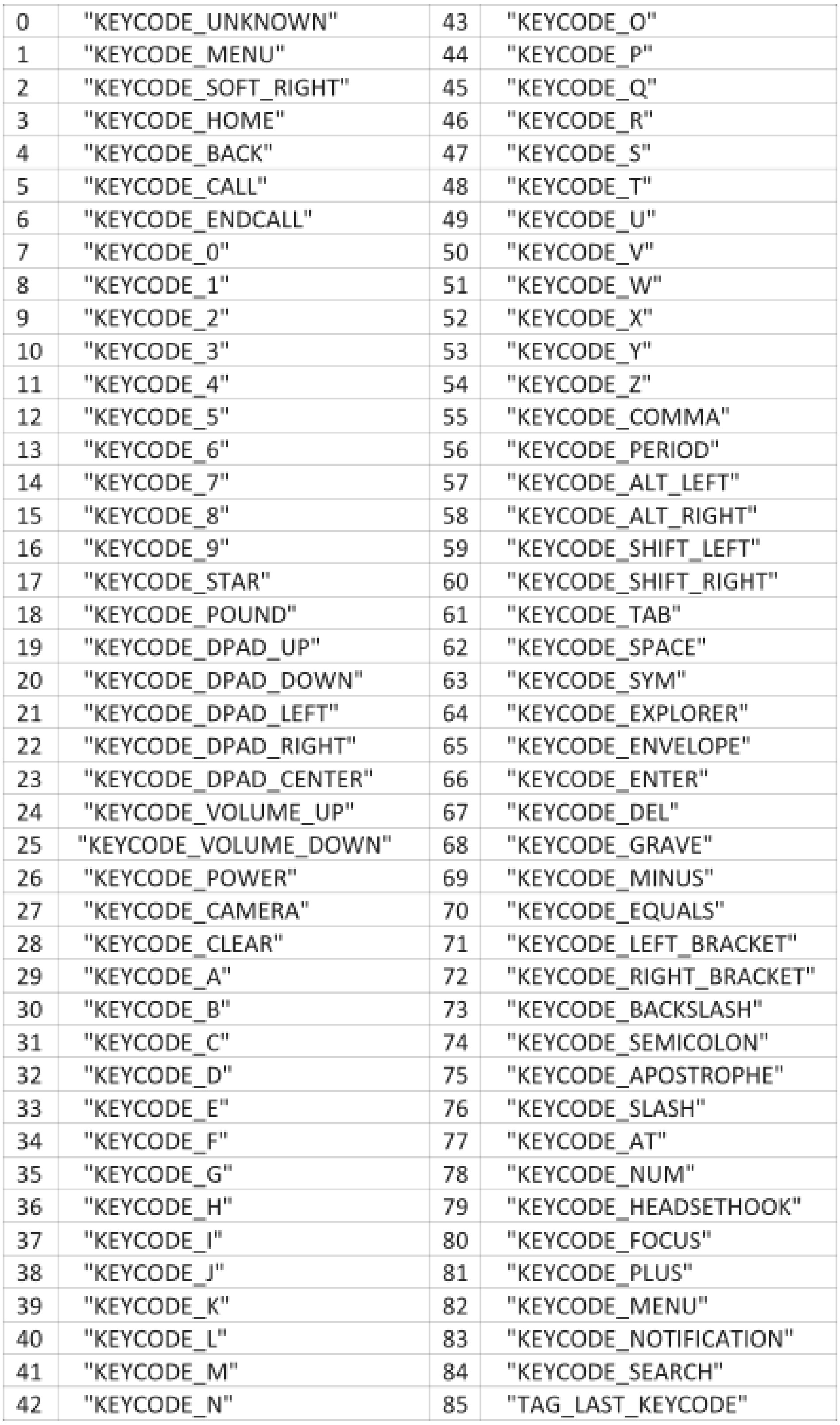
1. The h/w button triggers an interrupt.
2. The ISR of the corresponding driver gets called (in kernel).
3. In the ISR, the driver generates an input-event.
4. Android-framework (in userspace) gets the notification of the event.
5. Android reads the input-event code & type.
6. Compares it with the proper "keymap/keylayout" file.
7. The proper button is identified.

To simulate button presses we enter this procedure at STEP3.
Instead of the regular driver generating the input-event, we generate an input-event ourselves using a pre-built userspace binary. This will then be notified to the Android-framework and the rest continues as above.

So now all depends on the the pre-built userspace binary, which is...
[drum-roll...]
...**Input**. (How convenient!) The syntax of **input** is as follows**:**

```
input keyevent <event_code>
```

Now, before we try to send any keyevent, we need to find out the event-code that maps to the h/w key we want to simulate. The following table summarizes all the supported keycodes in Android:



0	"KEYCODE_UNKNOWN"	43	"KEYCODE_O"
1	"KEYCODE_MENU"	44	"KEYCODE_P"
2	"KEYCODE_SOFT_RIGHT"	45	"KEYCODE_Q"
3	"KEYCODE_HOME"	46	"KEYCODE_R"
4	"KEYCODE_BACK"	47	"KEYCODE_S"
5	"KEYCODE_CALL"	48	"KEYCODE_T"
6	"KEYCODE_ENDCALL"	49	"KEYCODE_U"
7	"KEYCODE_0"	50	"KEYCODE_V"
8	"KEYCODE_1"	51	"KEYCODE_W"
9	"KEYCODE_2"	52	"KEYCODE_X"
10	"KEYCODE_3"	53	"KEYCODE_Y"
11	"KEYCODE_4"	54	"KEYCODE_Z"
12	"KEYCODE_5"	55	"KEYCODE_COMMA"
13	"KEYCODE_6"	56	"KEYCODE_PERIOD"
14	"KEYCODE_7"	57	"KEYCODE_ALT_LEFT"
15	"KEYCODE_8"	58	"KEYCODE_ALT_RIGHT"
16	"KEYCODE_9"	59	"KEYCODE_SHIFT_LEFT"
17	"KEYCODE_STAR"	60	"KEYCODE_SHIFT_RIGHT"
18	"KEYCODE_POUND"	61	"KEYCODE_TAB"
19	"KEYCODE_DPAD_UP"	62	"KEYCODE_SPACE"
20	"KEYCODE_DPAD_DOWN"	63	"KEYCODE_SYM"
21	"KEYCODE_DPAD_LEFT"	64	"KEYCODE_EXPLORER"
22	"KEYCODE_DPAD_RIGHT"	65	"KEYCODE_ENVELOPE"
23	"KEYCODE_DPAD_CENTER"	66	"KEYCODE_ENTER"
24	"KEYCODE_VOLUME_UP"	67	"KEYCODE_DEL"
25	"KEYCODE_VOLUME_DOWN"	68	"KEYCODE_GRAVE"
26	"KEYCODE_POWER"	69	"KEYCODE_MINUS"
27	"KEYCODE_CAMERA"	70	"KEYCODE_EQUALS"
28	"KEYCODE_CLEAR"	71	"KEYCODE_LEFT_BRACKET"
29	"KEYCODE_A"	72	"KEYCODE_RIGHT_BRACKET"
30	"KEYCODE_B"	73	"KEYCODE_BACKSLASH"
31	"KEYCODE_C"	74	"KEYCODE_SEMICOLON"
32	"KEYCODE_D"	75	"KEYCODE_APOSTROPHE"
33	"KEYCODE_E"	76	"KEYCODE_SLASH"
34	"KEYCODE_F"	77	"KEYCODE_AT"
35	"KEYCODE_G"	78	"KEYCODE_NUM"
36	"KEYCODE_H"	79	"KEYCODE_HEADSETHOOK"
37	"KEYCODE_I"	80	"KEYCODE_FOCUS"
38	"KEYCODE_J"	81	"KEYCODE_PLUS"
39	"KEYCODE_K"	82	"KEYCODE_MENU"
40	"KEYCODE_L"	83	"KEYCODE_NOTIFICATION"
41	"KEYCODE_M"	84	"KEYCODE_SEARCH"
42	"KEYCODE_N"	85	"TAG_LAST_KEYCODE"

Additionally we can look up the keylayout (.kl) file on the device. Personally, I find keylayout a misnomer, as we are not talking about different keyboards here, but different mappings of an input-event-value to its functionality.

Anyways you can always find the file in **system/usr/keylayout/xyz.kl**
NOTE: For each keyboard device **xyz**, the **android.keylayout.xyz** system property must be set. If a keylayout file is not specified, Android will default

to**/system/usr/keylayout/qwerty.kl**

Now to generate a event of a specific keycode, we simply execute this on the terminal/serial console:

```
input keyevent <keycode>
```

The value of **keycode** can be any of the integer values from the above table. In case a serial-console is NOT available on your device, you can always run the command via adb as follows:

```
adb shell input keyevent <keycode>
```

That's how a hardware-button-press is simulated in software!!

Further Reading:

- More on keycodes, keymaps & Android-Input in general: Android/development/pdk/docs/keymaps_keyboard_input.html
- Android's Keyevent Class summary http://developer.android.com/reference/android/view/KeyEvent.html

66 comments :

tagmaster 12:16 am, April 03, 2011

Awesome! Are there any way to detect foreground App to send accurate key events?

Reply

Anonymous 4:35 am, April 18, 2011

Have no problem using KEYCODE_CALL but no luck to hang-up with KEYCODE_ENDCALL.

Reply

TheCodeArtist 7:46 am, April 18, 2011

@sj314 KEYCODE_ENDCALL CANNOT be intercepted by apps. Also since it is handled lower than the input queue, injecting it via adb may not work as expected.

More info here:
http://groups.google.com/group/android-platform/browse_thread/thread/c6e2de4b0cdbecd7/21b7cf4ee06a7005

Reply

takyon 11:54 pm, May 06, 2011

Is there a way to send a key press event for long press (The one that generally opens a context menu)

Reply

tagmaster 3:36 pm, May 25, 2011

what about COLON key?

Reply

TheCodeArtist 8:30 am, May 27, 2011

@tagmaster Any character/string literals can be sent directly using "input text" command.

For example to send a colon,

adb shell input text ":"

Reply

Anonymous 2:20 am, January 09, 2012

Hi,

still sending a text string with spaces is not supported.

Or at least not on my device
adb shell input text "a b c"
Will only print 'a' to the device screen

Workarounds are not so hard, just split the text on spaces and do 'adb shell input key SPACE' but it would be more convenient to have the adb to accept strings with spaces in them from the beginning

Cheers Johan

Reply

Replies

Apratim Utkarsh 12:10 pm, September 01, 2015

This comment has been removed by the author.

Reply

Anonymous 1:55 am, March 12, 2012

Can i simulate the power on event when the battery is off? I want my phone to power on as soon as a charger is connected . My specific requirement will not allow me to press power on button to start the android phone . Can any one has any idea whether this can be done? Iphone automatically starts itself if battery is drained off and put to charge . I want similar functionality in android phone. Please let me know if this is possible at all or not? Can we actually simulate power on button event when phone is off completely?Is it through alarm interrupt?if yes then details please.
Thanks in advance.

Reply

Replies

TheCodeArtist 1:25 pm, August 23, 2012

Short answer: No.

Long answer: The technique described in the above post requires Android to be booted-up on the device and adb setup properly between the device and the host-machine. Powering-on the device when charger is connected depends on how the bootloader is configured on it. When powered-off and connected to a charger, most Android devices just start charging and do NOT power-on until the power button is explicitly pressed.

Anonymous 5:08 am, October 14, 2013

Autobooting a tablet is very possible in fact but it requires modifications inside the kernel initramfs to hotswap the charger binaries.

Generally this means, compiling android from source (including a proper charging binary) and injecting that into the kernel as seen fit.

Have done it before. Worked fine.

TheCodeArtist 7:08 am, October 14, 2013

If we have access to the source, then the simplest way to get this working is to modify the boot-loader code. During boot-up the boot-loader checks if the power button was pressed, and if not then just keeps charging (often displaying a rudimentary charging UI) and does not load the Linux kernel.

Simply commenting the check and loading the Linux kernel unconditionally all the time should produce the desired behaviour.

Reply

Anonymous 3:38 am, August 30, 2012

The keylayout table you listed is working fine, but I tried to search for it on the system and didn't find it under /system/usr/keylayout in any "*.kl" file their. For example in your able the power keycode is 26 which is working on my device, but the power keycode in all my "*.kl" files are 116. I am confused.

Reply

▾ Replies

TheCodeArtist 12:06 pm, August 30, 2012

Here is the complete list of paths on the device where keylayout files can be stored.

Also the various paths listed in the above link are in order of priority. i.e. the system starts scanning for a .kl file in the path at the top of the list and continues down the list.

The first relevant .kl file found is used by the device.

Multiple .kl files can exist on a device. This is usually done to handle different input-devices like external-keyboards & joysticks. The input-device can generally be identified by the names of the .kl files.

Irrelevant keycodes can be listed in some .kl files. For example a joystick's .kl file could contain a (possibly wrong) keymap for the "power" button. Such codes do NOT affect the system behaviour as long as that particular input-device never generates that keycode.

It also could be done intentionally to provide similar functionality from multiple interfaces. For example, a particular button on the joystick can be configured to act as a power button for the Android device.

One can verify any .kl file using the Validate Keymaps tool (available as part of Android source code).

Reply

erocm123 1:09 am, October 13, 2012

This works great when I use terminal emulator, but when I log in via ssh or telnet and issue the same command (i.e. input keyevent 82), I get a "Segmentation Fault" error.

Anyway to run this with ssh and have the gui on the screen respond?

Thanks!

Reply

▾ Replies

TheCodeArtist 10:05 am, October 13, 2012

Not being able to run system commands is a common problem with various ssh servers on Android. It looks like its an issue with one or more environment vars not being setup properly which leads to missing lib and a seg-fault. You might want to run a trace on ssh from where you are trying to access the android device. Also the logs of the ssh-server on the Android device ought to be of help...

erocm123 10:53 pm, October 13, 2012

I did some more digging and found out that the version of su that I was using was causing the problem. It is the su that ships with the mk802 (which is what I am using). I downloaded Superuser from the market, and replaced the current su with it. Now it seems to be working, but it is a little slow. You don't happen to know of a way to speed it up do you?

TheCodeArtist 11:02 pm, October 13, 2012

Splendid job! :) Among other things, one way to speed-up ssh is switching to RSA keys which are almost 4times faster(for validation) than DSA.

Juan del Brio 4:16 pm, March 29, 2013

Hi erocm123,
How did you replace "su"?
I have tried replacing the su file but executing su I do not get the su prompt.
Can you provide the steps?
Thanks!!

Reply

Anonymous 11:30 am, November 03, 2012

can anybody tell me that if i want to do any action on long click then it is possible using adb shell ? yes then how ? because i had try out to find out but get only by click event . please help me .

Reply

Anonymous 5:15 pm, November 14, 2012

Thank you , helped me a lot :)

Reply

Naitse 3:32 am, December 07, 2012

is there a way to do the "opposite"

to hook a touch action to a hardware button, for instance

(I have first person shooters on tables because you cant aim and fire correctly)

in a game I press the screen to shoot which should be converted to a event, If I can attach that event to a button I'll be able to do the action with the button instead of the screen

the same way as if I attach a game controller to the tablet.

is that possible?

Reply

Anonymous 2:09 am, January 03, 2013

This works great! However , how can I use this to simuulate the screenshot capturing ? For capturing screenshot , *simultaneous* keypresses of volume down and power key are needed.

Reply

Anonymous 4:42 pm, February 27, 2013

i want to change the device date . changing date by 7 day ahead . i used intent function am start -a android.intent.action.MAIN -n com.android.settings/.DateTimeSettings
Now i can go ahead by 7 day by input keyevent to change the date .

Reply

▾ Replies

Anonymous 4:43 pm, February 27, 2013

*now i want to go ahead by 7 days by input keyevent to change the date

TheCodeArtist 5:25 pm, February 27, 2013

One can use the keyevents 19/20/21/22 (up/down/left/right) and 66(enter) to navigate the UI elements. The following sequence of keyevents does the trick for me on my N1 running CyanogenMod7.2(gingerbread):-

adb shell

```
input keyevent 20
input keyevent 20
input keyevent 66
(down,down,enter to select "set-date" option)
input keyevent 22
input keyevent 19
(move focus to "+" over day)
input keyevent 66
input keyevent 19
input keyevent 66
input keyevent 19
input keyevent 66
input keyevent 19
input keyevent 66
input keyevent 19
input keyevent 66
input keyevent 19
input keyevent 66
input keyevent 19
input keyevent 66
(7 times "enter". "Up" to refocus onto "+")
input keyevent 20
input keyevent 20
input keyevent 20
input keyevent 66
(move focus "down" to "Set" button and "enter")
```

This might have to change depending on your device and any UI modifications based on the ROM you are running. For example if "Set Date" option is further down in the list, then additional "keyevent 20"s will be required to navigate all the way down.

A simpler method for rooted devices would be to run the following command to set the required time and date.

```
date -s yyyyMMdd.hhmmss
```

Reply

Anonymous 8:36 pm, March 13, 2013

How do I delete say a list of characters for example.

The quick brown fox.

I could from the end of sentence execute the following command 20 times. But that executes rather slowly and it also seems superflous. Is there a execute multiple times command to make it simpler ?

```
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
input keyevent 67
```

Reply

TheCodeArtist 10:26 pm, March 13, 2013

Use shell scripts! ;-)

Either enter the entire set of adb commands in a shell-script on the host-PC and run it.

OR

Create a shell-script on the Android-device with all the "input keyevent keycode" commands and run this script on the Android-device via "adb shell script-name".

Reply

▾ Replies

Rhoadster91 7:25 am, July 17, 2013

Hi. I am creating a navigation bar alternative for Android and using these commands.

For back key, the command is input keyevent 4

But after the command is fired on older devices, it takes some time (about a second or so) for the response. That is rather slow.

Writing a shell script also is not slow as after every command, it pauses for a small amount of time.

Is there any alternative to speed this up?

Reply

Anonymous 6:27 pm, March 27, 2013

Thanks mate
this was useful

Reply

Anonymous 3:31 pm, April 29, 2013

Hi ,

I want to simulate Recent Apps icon on my Android device, can u guys pls let me know how to do it ....

Thanks in Advance......

Reply

monu 4:22 pm, May 22, 2013

I get the following error-
"Input cannot be resolved to a type"
Can anybody help??

Reply

Replies

TheCodeArtist 4:44 pm, May 22, 2013

What command did you run and where?...

monu 5:46 pm, May 22, 2013

I am a beginner and trying to write an android code to trigger the record button of the camera. I am launching the camera in video mode using intent and added -"Input keyevent 27" in my code to start recording but that produced an error "Input cannot be resolved to a type".

I then tried this-

```
KeyEvent kev = newKeyEvent(KeyEvent.ACTION_DOWN,KeyEvent.KEYCODE_CAMERA);
onKeyDown(KeyEvent.KEYCODE_CAMERA, kev);
```

but it didn't trigger the record button.
Please help. Thanks in advance.

TheCodeArtist 6:26 pm, May 22, 2013

Check out the tutorial to launch video recorder activity http://android-er.blogspot.tw/2011/04/start-video-recording-using.html

Also the following link has examples describing how to capture video in your own activity http://stackoverflow.com/questions/1817742/how-can-i-capture-a-video-recording-on-android

monu 6:30 pm, May 22, 2013

Can you please tell how to use "input keyevent" in android code!

TheCodeArtist 6:55 pm, May 22, 2013

Sending keyevents in java is accomplished by usingdispatchKeyEvent().

The documentation does go on to say that the keyevent is dispatched from the top of the view tree down to the currently focused view. If this view has focus, it will dispatch to itself. Otherwise it will dispatch the next node down the focus path. This method also fires any key listeners.

Reply

Anonymous 7:45 pm, June 08, 2013

Great info. I can not get this to work because my ICS tablet does not have the input method on it. I tried pulling it from the emulator, but it seems to have dependencies that I am not getting correct. Have you seen any good info on the correct installation of input and its dependencies?

Reply

Sergiu Strat 1:05 pm, June 11, 2013

Any idea how can I read Keyboard Inputs? For example I've pressed S key and I get X,Y coords for tap inputs and pressure in display area of that key... but nothing that could help me identify what key was pressed.

```
nevrax-McBkPr:~ nevrax$ adb shell getevent -lt
add device 1: /dev/input/event1
name: "lid_input"
add device 2: /dev/input/event0
name: "elan-touchscreen"
add device 3: /dev/input/event2
name: "gpio-keys"

[ 161550.717211] /dev/input/event0: EV_ABS ABS_MT_TRACKING_ID 000006ef
[ 161550.717228] /dev/input/event0: EV_ABS ABS_MT_TOUCH_MAJOR 00000006
[ 161550.717232] /dev/input/event0: EV_ABS ABS_MT_PRESSURE 0000000d
[ 161550.717237] /dev/input/event0: EV_ABS ABS_MT_POSITION_X 000000e3
[ 161550.717241] /dev/input/event0: EV_ABS ABS_MT_POSITION_Y 00000632
[ 161550.717247] /dev/input/event0: EV_SYN SYN_REPORT 00000000
[ 161550.862091] /dev/input/event0: EV_ABS ABS_MT_TRACKING_ID ffffffff
[ 161550.862104] /dev/input/event0: EV_SYN SYN_REPORT 00000000
```

Thank you!

Reply

▾ Replies

TheCodeArtist 1:16 am, June 12, 2013

The touchscreen driver is unaware of what is currently displayed on the screen. It just reports the X,Y co-ordinates.

If you are trying to setup some sort of automation/monitoring on a particular device that you own, then one possible way to determine the key pressed would be have a look-up table with all the co-ordinate ranges for the on-screen soft-keyboard that you have on your device.

It will be really challenging to regenerate the look-up table if the orientation of the device changes OR if the on-screen soft-keyboard layout is changed.

Sergiu Strat 3:33 pm, June 18, 2013

There are some tools that can record not only tap coords but also key inputs from keyboards and I was wondering how do they do it?

I'm trying to build an automation tool for repetitive tasks using a simple template system. In order to build those templates first I have to record my actions taps, swipes, key inputs and time between those actions. Otherwise I would have to enable debug mode an manually note each gesture and key input plus the time between actions.

TheCodeArtist 3:41 pm, June 18, 2013

Tools to record android keyboard input? Can you share any names/links to what you are referring to?

Sergiu Strat 4:11 pm, June 18, 2013

RepetiToch... but I think is having exactly the problem described by you... soft keyboard layout / changing screen orientation.

Android Bot Maker... seems not to be able to record anything but it can "play" taps and keys based on "input" command.

ArtOfMe 6:45 pm, February 21, 2014

did you guys happen to get this to work?
I use RepetiToch but that is time based and if I set it up the only way I think it will soon be out of sink with my automagic flow.
I tried to enable debug mode an manually note each gesture and key input plus the time between actions but it does not seem to do anything I get this in my logs.

```
21.02.2014 01:19:58.894 [Flow1] Action 'Execute Root Command: input tap in '
stderr=sh: [1]: syntax error: '<' unexpected filed
```

Reply

tang 3:12 pm, July 17, 2013

Great!

Reply

Anonymous 5:27 pm, July 23, 2013

so how could i type a german umlaut like öäü using "input keyevent"
using input text ö
leads to a segmentation fault.

Reply

▾ Replies

Unknown 11:49 pm, December 26, 2016

Hello. How did you fix input text for German umlaut?
Please send me your solution on oleh.prilepin@gmail.com
Thanks a lot

Reply

LatinSuD 6:23 pm, August 14, 2013

If you cannot use spaces with "text" is because a bug in /system/bin/input.

In that script you can replace:
$*
by
"$*"

or run the commands manually

Reply

▾ Replies

Unknown 11:36 pm, February 27, 2014

This comment has been removed by the author.

Unknown 11:39 pm, February 27, 2014

"$*" is not working for me. But "$@" is working true.

Reply

Anonymous 5:38 pm, August 21, 2013

Hello,

I tried to use the keyevent 26 (KEYCODE_POWER) to shut down the screen by a java application. It's don't seem to work neither for the KEYCODE_HOME but it's work s fine for KEYCODE_BACK... Is theirs some known issues for this problem ? thanks

Reply

▾ Replies

TheCodeArtist 6:37 pm, August 21, 2013

Exactly what commands are you running and how? On a PC connected to your Android device, does the following command from the console/command-line "adb shell input keyevent 26" work for you?...

Unknown 6:50 pm, August 21, 2013

When I use the adb shell input keyevent 26, it's works. But I would like to use it on a java file for the moment I tried :

Process chmod = Runtime.getRuntime().exec("input keyevent 26"); //it's only works for back pressed

&

Instrumentation mInst = new Instrumentation(); mInst.sendKeyDownUpSync(KeyEvent.KEYCODE_POWER); //it's only works for back pressed

TheCodeArtist 9:26 pm, August 21, 2013

The reason for the failure of "input keyevent" command launched using runtime.exec() is not clear to me as well. But i have NOT tried it and cannot vouch that it is supposed to work.

sendKeyDownUpSync() sends the event to the currently focused "window". Hence only the back button event works as the events generated by the power and home buttons generally are not handled by the foreground activity and sending the activities those events does not produce the desired effect.

Reply

Unknown 12:44 am, October 19, 2013

Any thought on how I use an external heat/motion sensor to trigger the camera on an android device?

Reply

Unknown 9:43 pm, October 21, 2013

Hello,
Is there a way to simulate a long power button press? I need to turn off the device with a long power button press. I've tried with adb shell input 26 but it seems like a short button press and my phone goes to sleep.

Reply

▾ Replies

TheCodeArtist 6:13 am, October 22, 2013

The long-button press is not handled in the software stack. That is why it works even when the Android OS is not-responding.

The power-button long-press event is generally implemented in hardware i.e as an interrupt to the PMIC (power-management chip). Upon receiving the interrupt, the PMIC momentarily cuts-off the power-supply to the main processor, effectively forcing a hard-reboot.

Use one of the following commands to shutdown/reboot your Android device :
adb shutdown -k *or*
adb reboot *or*
adb reboot -p

Reply

Unknown 12:06 am, October 24, 2013

adb shutdown -k seems to be not working. Sadly, I need to do a shutdown instead a reboot. Is there another way to shutdown without using the pwr button?

Reply

Unknown 6:28 pm, October 25, 2013

For your Query :

Sergiu StratJun 11, 2013, 1:05:00 PM
Any idea how can I read Keyboard Inputs? For example I've pressed S key and I get X,Y coords for tap inputs and pressure in display area of that key... but nothing that could help me identify what key was pressed.

```
nevrax-McBkPr:~ nevrax$ adb shell getevent -lt
add device 1: /dev/input/event1
name: "lid_input"
```

```
add device 2: /dev/input/event0
name: "elan-touchscreen"
add device 3: /dev/input/event2
name: "gpio-keys"

[ 161550.717211] /dev/input/event0: EV_ABS ABS_MT_TRACKING_ID 000006ef
[ 161550.717228] /dev/input/event0: EV_ABS ABS_MT_TOUCH_MAJOR 00000006
[ 161550.717232] /dev/input/event0: EV_ABS ABS_MT_PRESSURE 0000000d
[ 161550.717237] /dev/input/event0: EV_ABS ABS_MT_POSITION_X 000000e3
[ 161550.717241] /dev/input/event0: EV_ABS ABS_MT_POSITION_Y 00000632
[ 161550.717247] /dev/input/event0: EV_SYN SYN_REPORT 00000000
[ 161550.862091] /dev/input/event0: EV_ABS ABS_MT_TRACKING_ID ffffffff
[ 161550.862104] /dev/input/event0: EV_SYN SYN_REPORT 00000
```

Now we can change these hexadecimal code into Binary format , which actually ur system can understand .Below is the code which converts into binary one :

```
$LOGFILE = "play.txt";
$FILE = "Sample.pl";
open (CHECKBOOK, $LOGFILE);

while ($record = ) {
chomp($record);
if($record eq "sleep"){
print "sleep 1;";
sleep 1;
#continue;
}
@values = split(' ', $record);
$value0 = substr($values[0],0,17);
$value1 = hex($values[1]);
$value2 = hex($values[2]);
$value3 = hex($values[3]);
#if($value1 eq 0)
print "adb shell sendevent $value0 $value1 $value2 $value3 \n" ;



{system( "adb shell sendevent $value0 $value1 $value2 $value3 "); }

# foreach my $val (@values) {
# print "$val\n";
# }

}

close(CHECKBOOK);
```

Thanks
Shubham Tomar

Reply

Unknown 8:35 pm, April 02, 2014

nice article, just what I was lookin for.

Reply

Anonymous 3:44 pm, April 25, 2014

maybe anyone of You come with such a problem (accept button in camera app):
http://stackoverflow.com/a/23289891/2570350

Reply

Unknown 1:30 pm, July 09, 2014

m getting error when capturing KeyEvent_0.......KeyEvent_9. in receiver

Reply

Harvey 11:18 am, August 21, 2014

Thanks for the post! I was wondering if there is a way to press 2 keys at the same time. I am writing a command in tasker to help me login to to my wifi automatically. So far I have:

input keyevent 61; input keyevent 61; input keyevent 61; input keyevent 61; input keyevent 61; input keyevent 61; input keyevent 61; input keyevent 61; input keyevent 66

Basically 61 is tab which cycles me through the links on the login page until it stops on the "accept" button. 66 is enter which activates the "accept" button. This takes some time to cycle through and since the "accept" button is the last link on the page I would prefer to simulate Shift+Tab and then Enter. It would be much quicker. Thanks !

Reply

Noushad Sojib 5:13 am, October 11, 2014

Thanks for you'r nice explanation. It really helps me.

Reply

Unknown 2:53 am, November 03, 2017

Has anyone tried sending the keyevent codes to the android via an arduino? I am using the arduino to determine if the screen needs to be dimmed or if the tablet should go to sleep. This is a car head unit.

Thanks!

Reply

Unknown 6:10 pm, January 17, 2018

I have Honeywell ScanPal™ EDA50

How can I emulate the hardware button "SCAN" with the KeyEvent is KeyEvent.KEYCODE_UNKNOWN if I only know the scanCode.

Reply