

Share Your Experience: [Take the 2024 Developer Survey](#)



There's no 'sudo' command in Cygwin

Asked 14 years, 2 months ago Modified 2 years, 9 months ago Viewed 71k times



Because there's no [sudo](#) command in [Cygwin](#), scripts that I want to run fail with

44

`./install.sh: line N: sudo: command not found`



What's the standard way for getting around this? Editing the scripts to remove `sudo` ? Getting some `sudo` -like tool for Windows?



windows

command-line

cygwin

Share Edit Follow Flag

edited Sep 18, 2012 at 17:42



Peter Mortensen

12.2k 23 71 90

asked Mar 21, 2010 at 17:45



Jason Sundram

3,291 7 26 30



@dotancohen, I hope I picked the right one. – [Jason Sundram](#) Mar 30, 2015 at 0:16



Looks like a good solution to me! Cygwin support has certain gotten better in the past five years! – [dotancohen](#) Mar 30, 2015 at 7:35

1



See stackoverflow.com/questions/4090301/... – [Benj](#) Aug 3, 2016 at 9:29



@Benj, I wonder why that question didn't get migrated here, too. – [Jason Sundram](#) Aug 3, 2016 at 22:25

▲ @JasonSundram indeed. Let me know if the answer is moved, I'll then update the link. – Benj Aug 4, 2016 at 7:52
▼

9 Answers

Sorted by: Highest score (default) ▼



I wrote the (rather simple) [TOUACExt](#) for [SUDO for CygWin](#), a pre-beta shell script automation that approaches to the **behavior of classical** `sudo` for Linux:

10



- **Open and closes automatically** `sudoserver.py` when needed.
- Requests **UAC** Elevation Prompt.



Installation requires copying the four `.sh` scripts to some path directory, creating an alias and just a few more steps detailed in the thread.



The results: you type a single `sudo YourCommand` and you get the output of it, without having to worry about the rest of the process.

Share Edit Follow Flag

edited Mar 20, 2017 at 10:17

Community Bot
1

answered Apr 14, 2014 at 0:41

Sopalajo de Arrierez
6,723 13 66 99

One way is to create a fake "sudo" command with the following content:

36

```
#!/usr/bin/bash
```



```
"$@"
```



This will allow the `install.sh` to continue, because `sudo` is found.



This doesn't elevate privileges like real `sudo` does. If you really need elevated privileges start cygwin shell with from an account with administrative privileges (XP) or r-click on `cygwin.bat` and "*run as administrator*" (Vista, Win7)

Share Edit Follow Flag

edited Apr 12, 2011 at 7:13

answered Mar 21, 2010 at 18:17



matt wilkie

5,143 23 61 88



Peon

791 6 9

- 5 Just out of curiosity from someone who doesn't speak fluent `bash` : Why does this work? The manpage doesn't say anything about `$@` doing anything `sudo` -like. Instead it's just all arguments to the script. And wouldn't the quotes around it be superfluous in that case? Otherwise if you'd do a `sudo foo bar` then it tries executing `"foo bar"` as a single command which probably doesn't exist given that irrational fear of spaces on UNIX-like systems. – [Joey](#) Mar 23, 2010 at 10:29
- 8 @Johannes: `"$@"` (when double-quoted) works differently from `"$*"` : it expands to a **separate** word for every positional variable. **Example:** If `$1 == "foo bar"` and `$2 == "baz"` , then `"$@"` is `"foo bar" baz` - one word for each parameter (unlike `"$*"` , which results in `"foo bar baz"` as one word). See manual of `bash` , section *Parameters*, subsection *Special parameters*. The end result of Peon's script is that it executes its arguments exactly as they were passed. – [gravity_u1686](#) Mar 23, 2010 at 15:22
- 1 Ah, ok. And where does the `sudo` part come in? Above snippet doesn't do anything remotely in that direction, right? – [Joey](#) Mar 23, 2010 at 15:38
- 2 @Johannes: In Unix, a real `sudo` would raise privileges from mortal to `root` before running the command. In Cygwin, there is no such thing, so Peon's fake script (which you're supposed to name `sudo`) just runs the command directly without changing its privileges. (This means you may need to run `./install.sh` as Administrator.) – [gravity_u1686](#) Mar 23, 2010 at 15:49
- 2 @gravity: `runas` should work, it doesn't rely on UAC and prompts for a password by itself. I was just confused why the script in the answer apparently didn't do what the name implied which I assumed was the goal. Sorry for my stupidity ;-) – [Joey](#) Mar 23, 2010 at 17:12



29



I found the answer on [the cygwin mailing list](#). To run `command` with elevated privileges in Cygwin, precede the command with `cygstart --action=runas` like this:

```
$ cygstart --action=runas command
```

This will open a Windows dialogue box asking for the Admin password and run the command if the proper password is entered.

This is easily scripted, so long as `~/bin` is in your path:

```
$ cat ~/bin/sudo
#!/usr/bin/bash
```

```
cygstart --action=runas "$@"
```

```
$ PATH=$HOME/bin:$PATH  
$ chmod +x ~/bin/sudo  
$ sudo elevatedCommand
```

Tested on 64-bit Windows 8.

Share Edit Follow Flag

edited Jan 9, 2014 at 15:34

answered Jan 9, 2014 at 15:25



dotancohen

11.5k 19 67 98

7 ▲ The problem with the `cygstart` method is that it only works for Windows commands/programs. You can not do `sudo ls`. SUDO for CygWin is neat, but still lacks some good `sudo` command in it. – [Sopalajo de Arrierez](#) Apr 13, 2014 at 20:59

▲ Thank you, Sopalajode. In what situation have you needed to use `sudo ls` in Cygwin? – [dotancohen](#) Apr 17, 2014 at 10:04

3 ▲ Oh, no, @Dotancohen, it was just an example. You can use `sudo` for CygWin to run any Windows or CygWin command. It is very useful for me. But the more practical method I have found is this script wrapper for SUDO for CygWin I developed: superuser.com/questions/741345/... (still in Beta, but seems to work). With it you can comfortably order things like `sudo net start vncserver` . – [Sopalajo de Arrierez](#) Apr 17, 2014 at 10:22

▲ @SopalajodeArrierez: That is absolutely terrific! Thank you for the post and the link. – [dotancohen](#) Apr 17, 2014 at 12:15

1 ▲ tried almost everyother solution. this is the only one that worked – [rajesh](#) Jun 9, 2020 at 11:40

|



Building on [dotancohen's answer](#) I'm using an alias:

5

```
alias sudo="cygstart --action=runas"
```



Works as a charm for external programs (not shell built-ins, though):





`sudo chown User:Group <file>`

Share Edit Follow Flag

edited Mar 20, 2017 at 10:04



Community Bot

1

answered Jan 16, 2016 at 12:58



thoni56

185 1 14



Sudo (Elevate) for Windows™

5

I do a lot of work on the command line in Windows™.



In Cygwin itself I believe you can run a root command with `su -c /the/cmd` as for sudo itself within Windows™ file-system elevating the user's permissions from the command line, If you are an administrator, this will work great for you. Otherwise, use runas and get admin's pass ;).



Now I cannot remember where we got this code but here it is. I hope it helps.

BTW, the package we use to compile this was `gcc-mingw32`.

```

    }
}
else
    result = GetLastError();

if (result != 0x2a) {
    LPWSTR errmsg = NULL;
    FormatMessage(FORMAT_MESSAGE_FROM_SYSTEM | FORMAT_MESSAGE_ALLOCATE_BUFFER,
        NULL, result, 0, (LPWSTR)&errmsg, 0, NULL);
    MessageBox(NULL, errmsg, TEXT("Error:"), MB_OK | MB_ICONERROR);
    LocalFree( errmsg );
    return result;
}
else
    return NO_ERROR;
}

```

Share Edit Follow Flag

edited Sep 18, 2012 at 17:53

answered Feb 13, 2012 at 16:16



Peter Mortensen

12.2k 23 71 90



tao

1,415 8 11

- 6 ▲ This code is awful. It's full of errors like not checking the return value of `realloc()` before dereferencing or writing `sizeof(LPWSTR*)` instead of `sizeof(*LPWSTR)` where `LPWSTR` seems to be a pointer type and one wants to retrieve the size of one character, not the size of the pointer. Furthermore, it's totally unclear why e.g. case 29 leads to `ERROR_DDE_FAIL`. Can you conclude from the code why? I can't and I guess anybody else can't, too. Please, do not post such code in the future. – user185282 Jan 8, 2013 at 8:59
- 4 ▲ @Matthew: please in the future help the community by posting a cleaned up version of the code snippet you dislike. – Erik Kaplun Jan 20, 2013 at 14:36
- ▲ Code shouldn't be on superuser. Put it on [codereview.se](#), and just link it from here. – Ben Voigt Apr 14, 2014 at 2:19
- ▲ @user185282: Good points. I've downvoted the answer. – [unforgettableidSupportsMonica](#) Mar 20, 2017 at 3:28
- ▲ Dear tao: You wrote, "I cannot remember where we got this code". Did you write this code, or did someone else write it? – [unforgettableidSupportsMonica](#) Mar 20, 2017 at 3:28

A slight improvement on Peon's *fake sudo* script:

3

```
#!/bin/sh
# Drop any option arguments.
while [[ $# -ge 0 && $1 = -* ]]; do
    shift
done

"$@"
```

This script silently drops any options passed to sudo and executes the command (without actually elevating the privileges). Dropping the options improves compatibility somewhat. A more complete wrapper script should actually parse options the same way sudo does.

Instead of trying to replace sudo with a wrapper that does `cygstart --action=runas "$@"`, just use this simple *fake sudo* wrapper and run your install script itself (or whatever you're trying to run that uses sudo) with elevated privileges.

Share Edit Follow Flag

answered Apr 20, 2016 at 15:18



Gene Pavlovsky

533 5 12

▲

In scripts which can run on linux or windows (cygwin) you can alias out sudo in the script itself:

0

```
# Alias sudo to do nothing on windows
if ! $(type -P sudo)
then
    shopt -s expand_aliases
    alias sudo=""
fi
```

▼

🔖

Share Edit Follow Flag

answered Apr 26, 2021 at 11:31



meesern

131 1 4

▲

I've been using [cygwin-sudo](#) as a drop-in replacement for UNIX/Linux's sudo.

0 Based on the same principals but one can remain within your current window.



Share Edit Follow Flag



answered Mar 31, 2021 at 17:35



phdye

1 2



You can run sudo but may require options. If you know the user/group/ Active Dir environment you can supply those as options.

1

Here is a quick alias or you can "sudo" shell script in common PATH as previous example.



```
alias sudo='runas /noprofile /user:Administrator $@'
alias sudo-='runas /profile /env /user:Administrator $@'
```



Before you can use "sudo" you need a running/enabled windows service called "seclogon".

```
cmd /c 'sc config seclogon start=auto'
cmd /c 'sc start seclogon'
```

or use powershell in following link: [How do I enable services on a Windows computer via a command line?](#)

Share Edit Follow Flag

answered Aug 29, 2021 at 15:36



PaSe

11 1