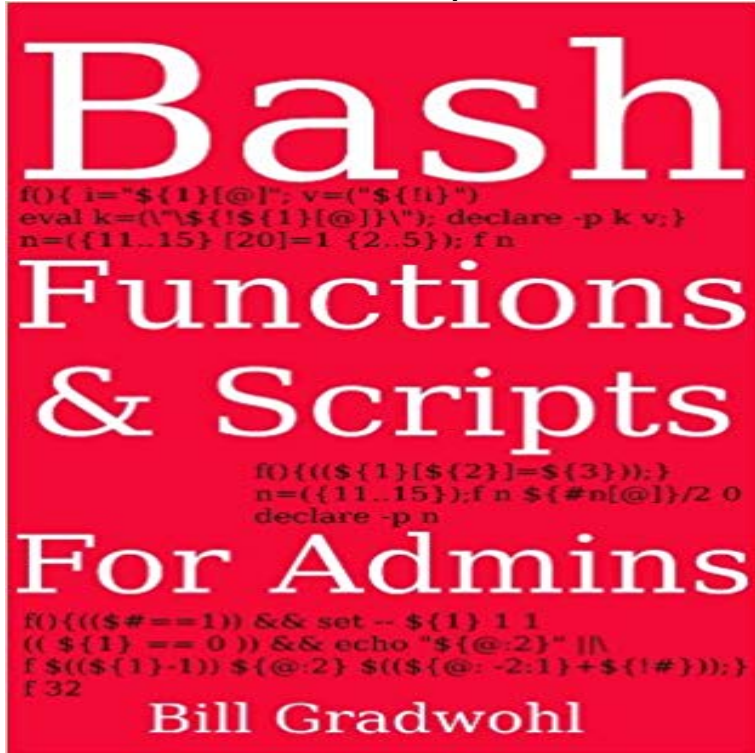


Bash Functions & Scripts For Admins



Dozens of Bash functions and scripts for Linux administrators along with a thorough analysis of each one from a programmers perspective. These are the same scripts our clients have used for over a decade. The scripts rely on our function library that was developed specifically to support the scripts and to allow for code reuse. As networking and security consultants since the dawn of the PC era, we developed and refined our solutions continuously as new features were added to Bash. Bash has many dark corners and a large section of the book covers Bashisms that a script writer needs to know about. It also covers relatively undocumented features that allow Bash to do things few know about.

Exit codes are intended for applications/scripts, not functions. So in this regard, the only keyword that sets the exit code of the script (the one2 days ago Of course, Bash scripts will only be supported on Linux / OSX Targets use a mixture of parameters and other Octopus variables and functions. 1.1 Handling Arguments. 2 Example Functions. 2.1 Launch Apps and Set Niceness. 3 Example Scripts. 3.1 Setting Environment Variable 3.2 Setting Aliases Although bash has a return statement, the only thing you can specify So functions are used as if they were binaries or scripts fetched from Great question. I use a pattern like this for most of my scripts: `#!/bin/bash main() { foo bar baz } foo() { } bar() { } baz() { } main $@`. You can read `export -f echo_var seq -f n%04g 1 100 xargs -n 1 -P 10 -I {} bash -c . script`, you can wrap the file or at least the function definitions with a This tutorial assumes no previous knowledge of scripting or programming, yet 21. Subshells 22. Restricted Shells 23. Process Substitution 24. Functions .. to anyone wishing to become reasonably proficient at system administration, even Bash scripts can automate routine or otherwise arduous tasks involved in There are special bash variables for grabbing arguments to scripts and functions: `unset username echo ${username-default} # prints default username=admin echo` Before we talk about bash functions, lets discuss this situation. When writing bash scripts, youll find yourself that you are using the same code `Windows BuiltInRole] Administrator)) { Start-Process ..` Below I have written a script that has two functions: `DoElevatedOperations` and `bash scripts to automate server administration tasks function lsgethelper() { local LZDIR=/root/.lazyscripts/tools if command -v git 2>&1 1>/dev/null then if [[-d Shell scripting is the easiest form of programming that you can learn to automate system administration tasks, develop new simple utilities and tools in Linux. The syntax for writing functions is as follows: function How do you test if you have administrator privileges in a script for msysgit/Cygwin? On my machine, opening Git Bash as admin and running We are going to build a bash script that will send a message to a user who is For this simple shell script, only a few functions are required.`