How to use alias

Aliases give us a great possibility to "shape" command line to our needs. Let's suppose, we want to list all files in long format and with hidden files:

ls -al

We have to use 6 characters. not much. But if we need to repeat this command again and again, it may become tough. So, what about this:

ll

Better?

Definitelly.

ll (and a few more) is an alias configured in our profile configuration (please have in mind, not always, not for all users, not in all systems). Before we see how to do it, let's create some aliases in command line first.

In order to create alias we need to use alias command. This time it is not an abbreviation :). Ok, let's create an alias for ls with long list, hidden files and human-readable format of size.

alias lh='ls -alh'

What we did?

We want to create an alias called lh and when we invoke it, it should execute the ls -alh command.

Now we are redy to execute this alias:

lh

Please remember, this alias is available in this session **only**. It is not available in other sessions, it will be not available when new session is started.

Ok, are you curious what aliases you have in your system? We have lh now, we know about ll. Is there anything else?

alias shows all defined aliases in the session.

Sometimes we want to remove aliases. Well, honestly, I never needed to do that, only during the trainings :)

anyway, lert's remove the alias created during this lab:

unalias lh

Can we remove alias, if it was defined earlier, somewhere in the configuration? Let's try:

unalias ll

Ok, now we can check the alias

And let's try ll, what we should see?

ll

Make alias permanent

We have at least two ways of creating aliases "forever".

First, take a look on .bashrc file.

grep "alias " .bashrc

As we can see, there are some entries for aliases. Let's add our previous one to .bashrc. We don't care about the strict location in the file for now. We'll add it on the end.

echo "alias lh='ls -alh'" >> .bashrc

Let's see:

grep "alias " .bashrc

In order to use this alias we need to load .bashrc into our session.

source ./.bashrc

source is very useful in many cases. It reads and executes scripts (or should we say - content of the files, as it doesn't need to be script) in **current** shell. By default, when we execute a script, it will be executed in its own shell. We will come back to this context soon.

Check the alias:

lh

Another file, where we can add our alias is .bash\_aliases. This file should be loaded automatically when .bashrc is loaded.

In order to be sure, we need to check if the proper block is defined in .bashrc:

# Alias definitions.

# You may want to put all your additions into a separate file like

# ~/.bash\_aliases, instead of adding them here directly.

# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash\_aliases ]; then

. ~/.bash\_aliases

fi

We can check if we have the file already define, if not we should create it. But with redirections we are able to add something to file (if file exists), or add new file. Let's use it.

echo "alias lh1='ls -alh'" >> .bash\_aliases

We can list our file:

cat .bash\_aliases

And load it to our current session:

source ./.bashrc

Please, notice that we didn't load the .bash\_aliases, but .bashrc. Just a few lines above we have the if statement in order to load additional file.

Let's check our aliases:

alias

And execute the new alias:

lh1

# Aliases for all users

Here we touch the administrative part.

To this point we created aliases for current user. We are able to create aliases for all users. In order to do this, we have to add something to global configuration.

There are couple of ways how to do it, however we will learn the best one.

Here in Killercoda we are logged as root, so nothing additional is needed to do.

The place where we have to add our aliases is /etc/profile.d.

A little theory. When you log in to the system, the ~/.profile file is loaded. But before this file system loads the main and common configuration from /etc/profile file. This file does different things and loads all files from /etc/profile.d directory. Files are loaded in alphabetical order, so one of the good practices is to keep numbers in the files, if the order is important for us. So, let's supose, our aliases are not that important, therefor we can load it on the very end of this execution.

Let's create the file, then, and write another alias.

echo "alias lh2='ls -alh'" >> /etc/profile.d/99-aliases.sh

As you noticed, we saved it as .sh file.

We can add this alias to the /etc/profile file directly, but it is strongly not recommended.

Let's see if it works. First, we will start new session - sudo -i

(we will touch it in one of the next lab) and what we see after the

alias execution?