Building CDP Kernel Module

R1Soft does not provide pre-built modules for the popular kernels anymore, so you will have to compile the module from source. You will have to compile this module against kernel headers or a kernel source tree.

CDP Device Driver is a proprietary, loadable Linux kernel module distributed by R1Soft. It is loadable at run-time without restarting Linux, and you do not need to recompile your Linux kernel to use it.

> ⚠ Notice
> You need to have loadable modules enabled as a feature in your kernel, and this is standard on all popular Linux distributions.

If the kernel devel is too old and not found, see how to setup access to older `yum` packages in http://wiki.r1soft.com/display/kb3/Finding+Old+kernel-devel+Packages+For+CentOS.

1. Compiling CDP Kernel Module Against Kernel Headers or Kernel Source Tree

We are not always able to compile kernel modules from kernel-devel packages supplied by most major Linux distributions. In some cases, packages are missing header files (broken), or the packages have been stripped of information that any device driver would need to compile a kernel module. In these cases, we can build using an installed `kernel-devel` package on your Linux server, as `r1soft-setup` will obtain the missing information it needs to compile a module from your running kernel.

In order for kernel module compilation to work, you should have Internet connectivity directly from the Linux server you are installing CDP on, to TCP port HTTPS (443), on the host [krnlbld.r1soft.com].

You can test connectivity with the following command (this may take a minute):

```
# r1soft-setup --test-connection
```

2. Installing Kernel Sources

If you are using an unmodified kernel provided by CentOS installer, install the `kernel-devel` package:

```
# yum install kernel-devel
```

```
[root@centos-server ~]# yum install kernel-devel
Loaded plugins: fastestmirror
Determining fastest mirrors
 * base: mirror.karneval.cz
 * extras: mirror.karneval.cz
 * updates: mirror.karneval.cz
base                                                      | 3.7 kB     00:00
extras                                                    | 3.5 kB     00:00
r1soft                                                    |  951 B     00:00
updates                                                   | 3.5 kB     00:00
updates/primary_db                                        | 2.2 MB     00:12
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package kernel-devel.x86_64 0:2.6.32-220.7.1.el6 set to be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch                Version                  Repository              Size
================================================================================
Installing:
 kernel-devel           x86_64              2.6.32-220.7.1.el6       updates                 7.3 M

Transaction Summary
================================================================================
Install       1 Package(s)
Upgrade       0 Package(s)

Total download size: 7.3 M
Installed size: 23 M
Is this ok [y/N]:
```

3. Verifying that the Source Matches Your Running Kernel

Sometimes, the kernel-devel package is newer than the installed and running kernel. If the kernel-devel is too old and not found, please follow the instructions on how to setup access to older yum packages as documented here.

4. Building the CDP Kernel Module

Option 1: Building Online

(Direct Internet connection to R1Soft build server).

To attempt to build the kernel module, run the following command (this may take several minutes):

```
# r1soft-setup --get-module
```

If module has been compiled and installed successfully, you will see an output similar to the following:

```
Saving kernel module to '/lib/modules/r1soft/hcpdriver-cki-2.6.32-220.4.1.el6.x86_64.ko'
Kernel module is now installed.
Use '/etc/init.d/cdp-agent restart' to load the new driver
```

Option 2: Building Offline

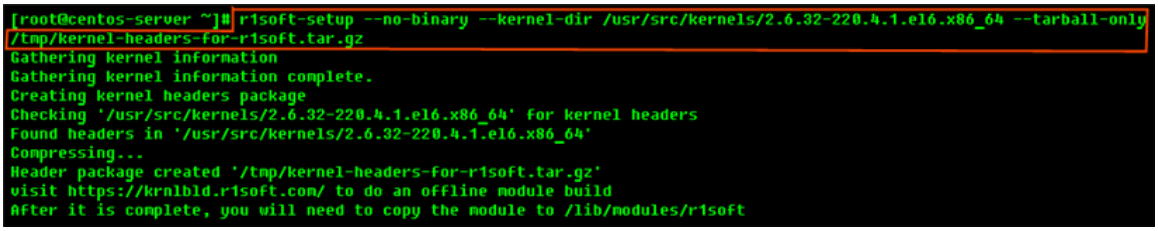
(Without direct Internet connection to R1Soft build server).

If there is no direct Internet connection between your CDP server and R1Soft build server, it is still possible to compile the kernel module. In this case, this will be tree-step process. First, you will have to create tarball file with the kernel headers. When you should copy this tarball file from the server to some other computer which has Internet connection to R1Soft build server. From this computer you should upload the tarball to the R1Soft build server and wait for the compilation to finish. When it is finished, you should download binary module and copy it back to the CDP server. Start with executing the following command:

```
# r1soft-setup --no-binary --kernel-dir /usr/src/kernels/YOUR_KERNEL_TREE
--tarball-only /tmp/kernel-headers-for-r1soft.tar.gz
```

After running this command, you will see:

```
# r1soft-setup --no-binary --kernel-dir /usr/src/kernels/2.6.32-220.4.1.el6.x86_64
--tarball-only /tmp/kernel-headers-for-r1soft.tar.gz
Gathering kernel information
Gathering kernel information complete.
Creating kernel headers package
Checking '/usr/src/kernels/2.6.32-220.4.1.el6.x86_64' for kernel headers
Found headers in '/usr/src/kernels/2.6.32-220.4.1.el6.x86_64'
Compressing...
Header package created '/tmp/kernel-headers-for-r1soft.tar.gz'
visit https://krnlbld.r1soft.com/ to do an offline module build
After it is complete, you will need to copy the module to /lib/modules/r1soft
```



When the tarball (`.tar.gz`) file has been written to disk, perform the following steps:

- Copy the generated `tar.gz` file and paste it to a computer with Internet access.
- Go to https://krnlbld.r1soft.com/ and upload the `.tar.gz` file to build a kernel module.
- After the build, you will download a kernel module.
- Copy this module and paste it to your Linux Server and the folder `/lib/modules/r1soft.`
- Restart the Agent (`/etc/init.d/cdp-agent restart`).