v1.0.0

VPAND.COM

Virtual Processor Infrastructure for Code Reverse Engineer and VMProtect.

Loading very hard for Qt.wasm, wait a second please...

Powered by LLVM/Clang.

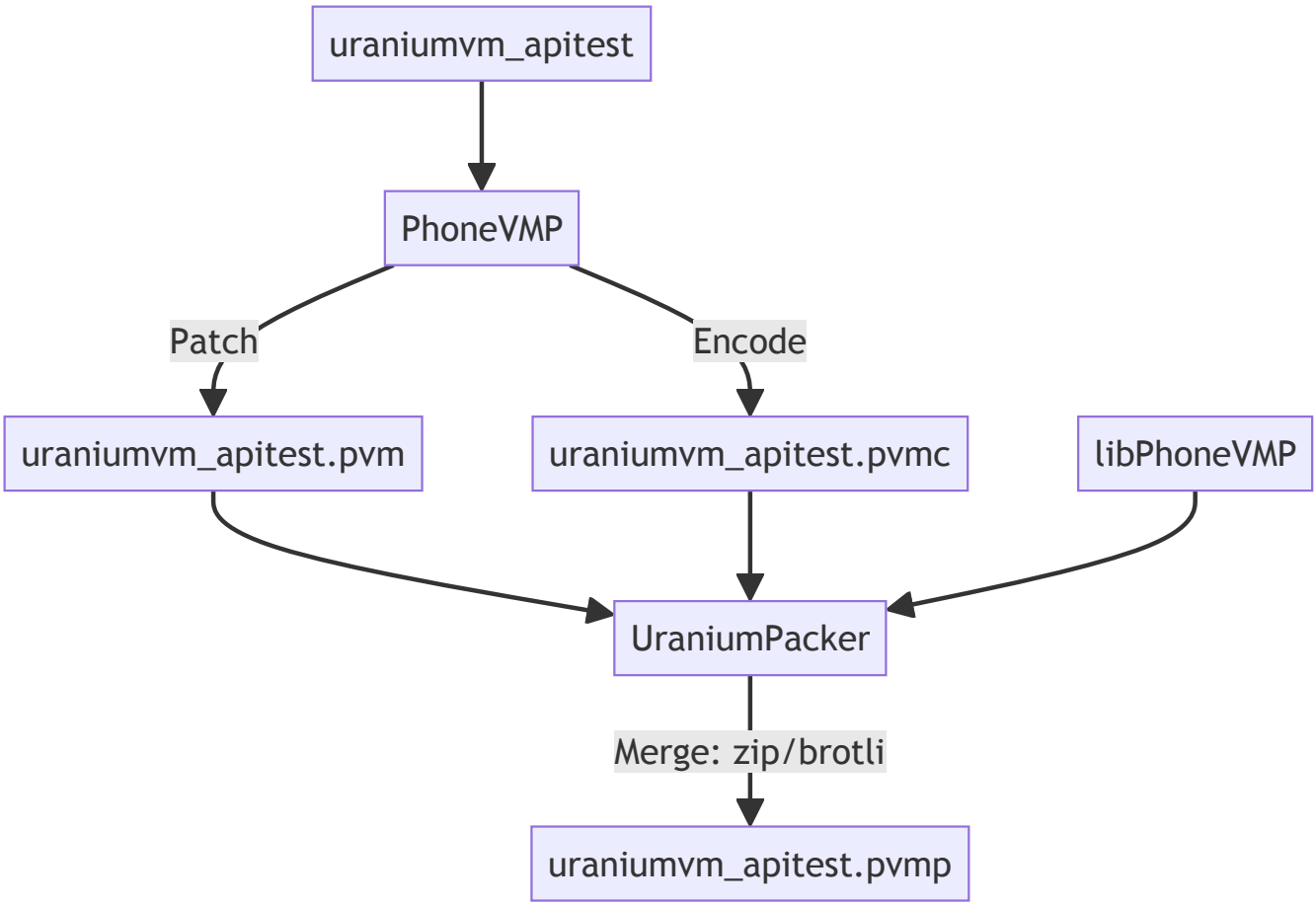
UraniumPacker User Manual

UraniumPacker

UrPacker (Uranium Packer) is an **arm/arm64/x86/x86_64** binary shell for Darwin (**macOS/iOS**) and **Android** operating system. UrPacker is not only a **compression** shell, but also a MachO/ELF **merging** shell, which bundles multiple binary files into one, and then decompresses and links them into memory at runtime by UrPacker Runtime Library, similar to the UPX decompression process.

Multiple file merging is one of UrPacker's core values, so what scenarios require such merging shell? For example, if you want to hide the actual binary file to resist the analysis of some people, then one more memory dumping and fixing steps can dissuade some people to do the real reverse engineer. Further more, if combined with other custom anti-dynamic analysis schemes, it's not that easy to dump.

For another example, your binary file depends on a number of scattered files, and their dependencies will expose your design architecture and purpose, combined together then that is a black box, it's difficult to clarify the relationship between them. After all, **debugging memory module is much more difficult than the normal file module**. The resulting pvmp file in the following figure is created by UrPacker by combining three files: pvm, pvmc, and libPhoneVMP.



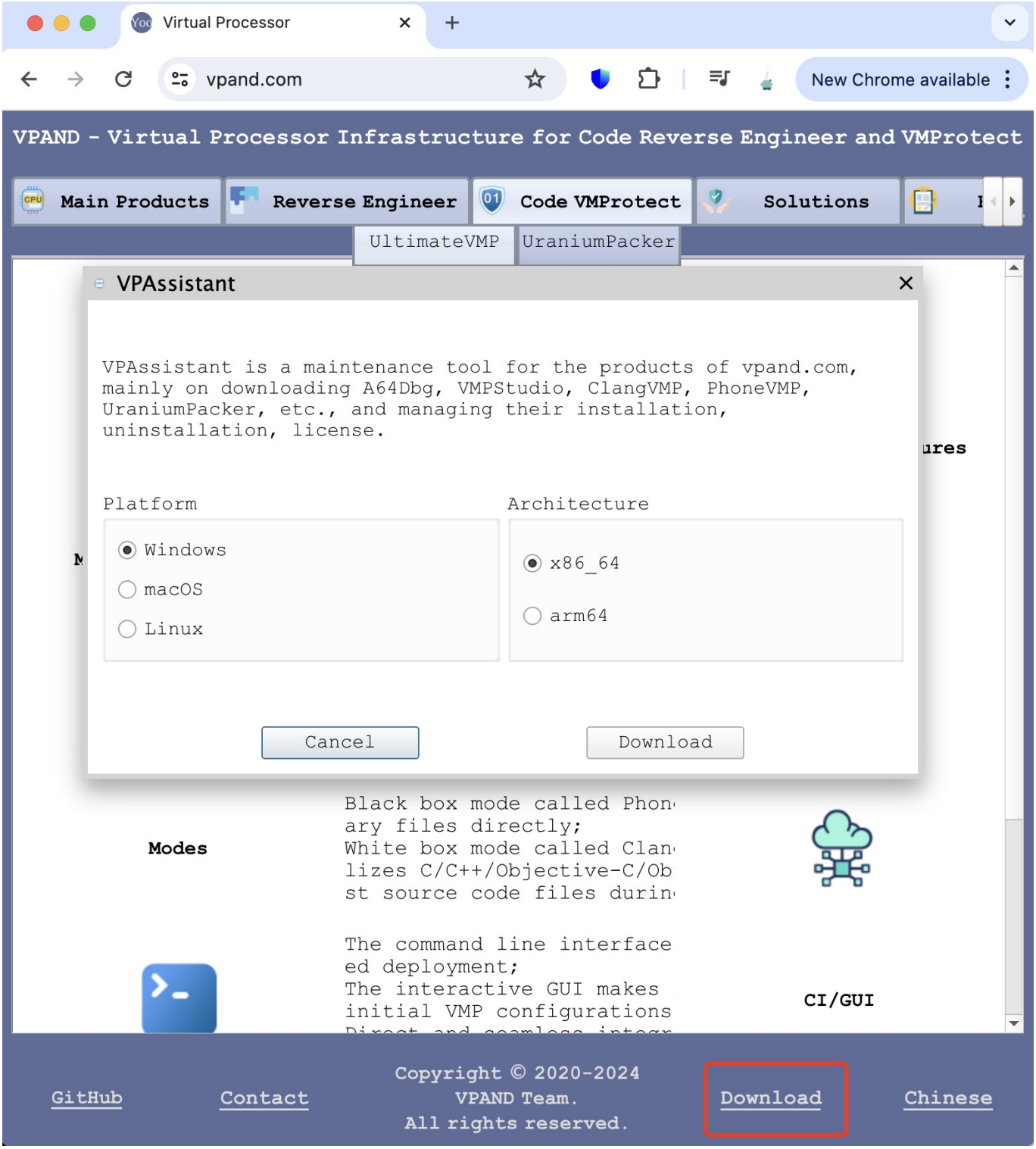
Compression is another core value of UrPacker. This is usually achieved by using higher compression rate algorithm like Brotli. As we know, product package on iOS and Android is a zip file, but the zip compression rate is much lower than brotli, so if we **pre-compress huge library file** like chromium with

UrPacker, then the final product package will be smaller which will have less download time that is absolutely good for your new customer. Because sometimes when the package is too big, the customer may have no patience to wait for finishing download.

Download

All of our products are hosted in vpand.com, you can download the assistant tool called **VPAssistant** to fetch your interested product like UraniumPacker.

Please note that you should download **the exact platform and architecture** which matches your current running OS, all the real product download through VPAssistant highly depends on the architecture that it's running on. For the native performance, you'd better not download x86_64 version on arm64 macOS even though it can directly run on it with Rosetta 2.



Install

As the VPAssistant on different platform(like Windows, Linux and macOS) is **absolutely the same**, unless some feature is just available on that specified platform, otherwise all the generic feature screenshots are from macOS. Here we go.

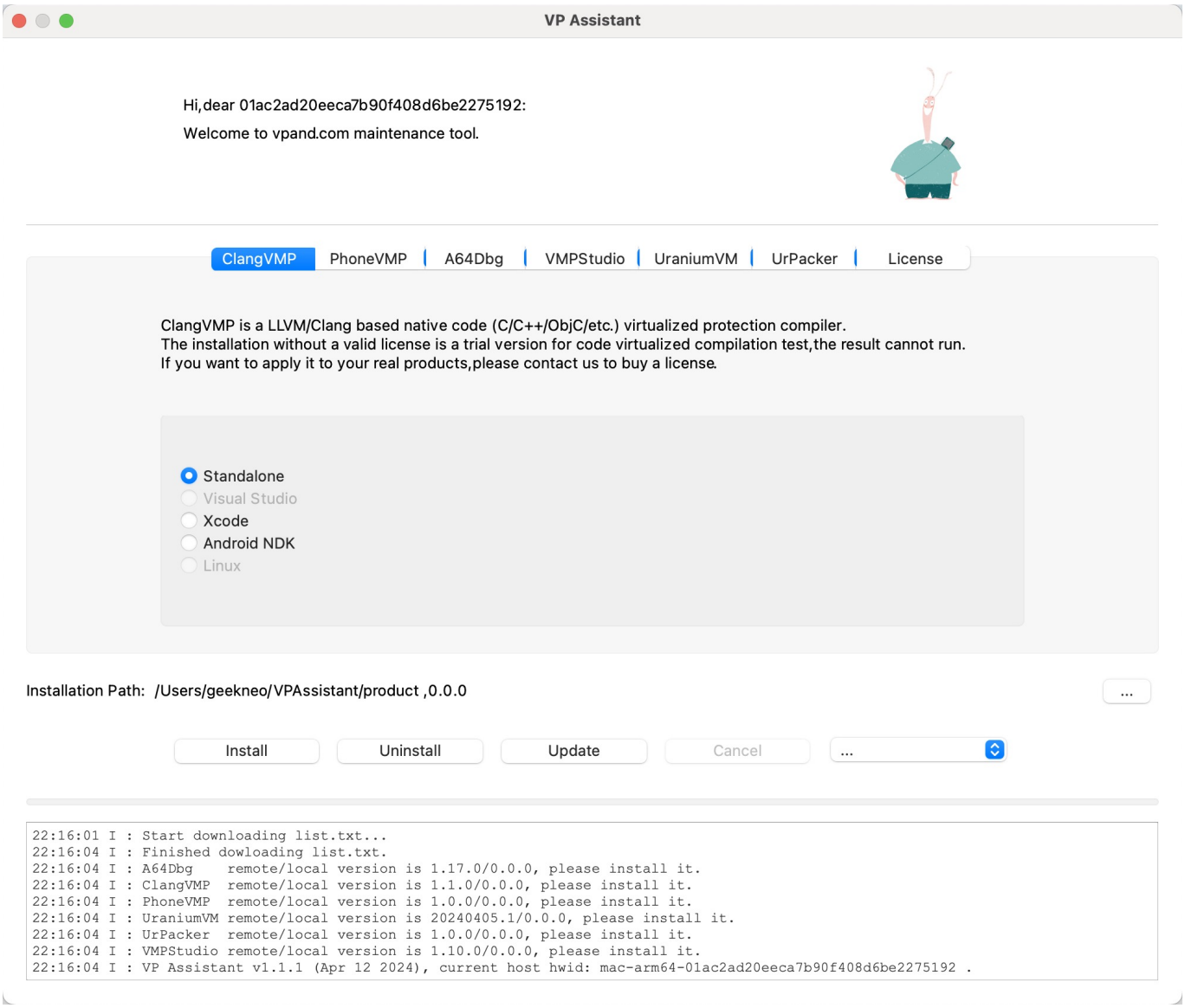
After download, unzip and launch VPAssistant, we're gonna be in the ClangVMP tab widget, then click the **UrPacker** tab to activate the assistant for UrPacker. The default installation path is(You can **change it with the "..." button** on the right):

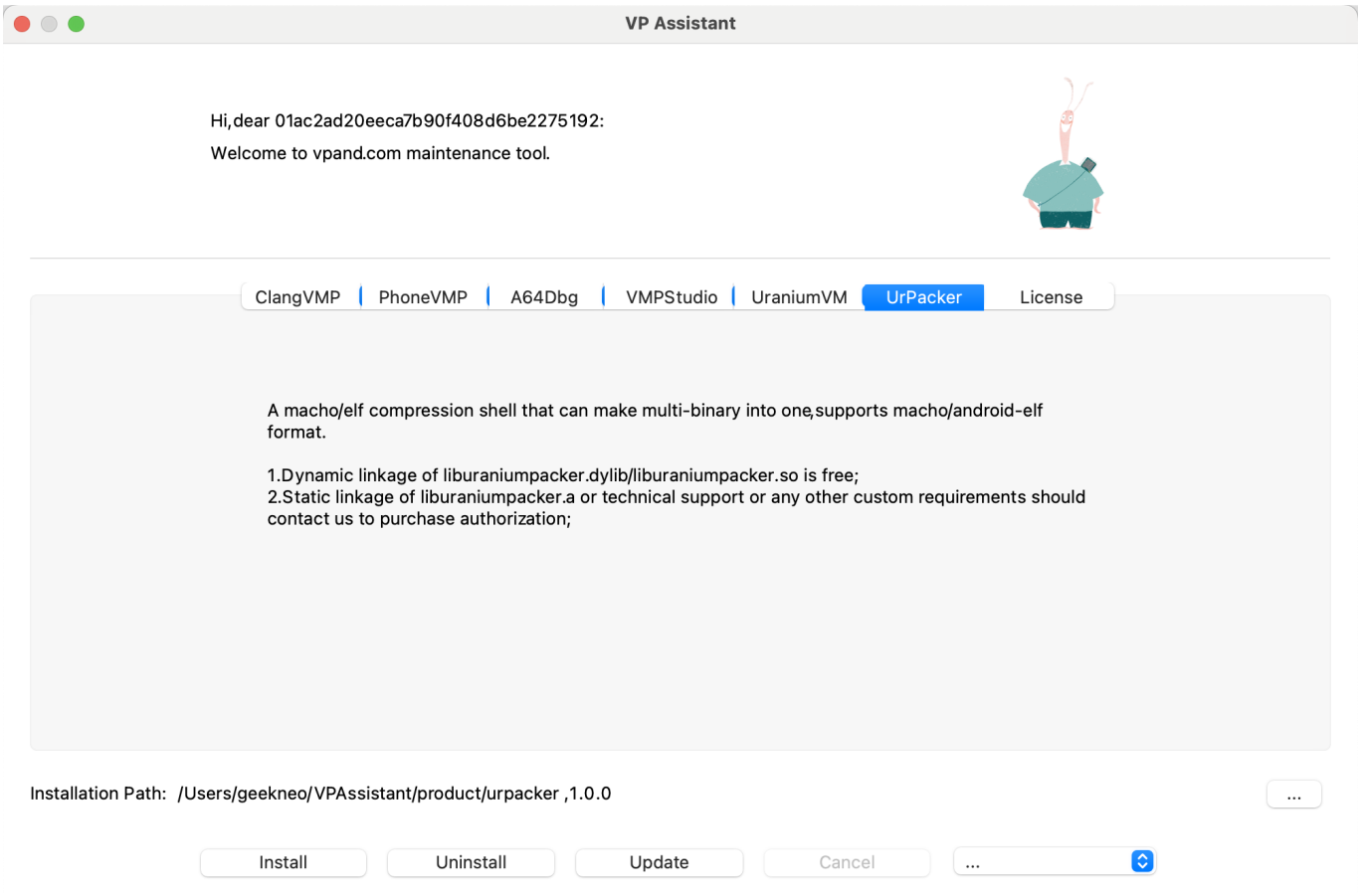
```
macOS/Linux : ~/VPAssistant/product

Windows     : SysDrive:\Users\user-name\VPAssistant\product
```

Every time when VPAssistant finishes launching, at the end of logs there'll be a line for your running OS triple: os-arch-hwid. It's the key to authenticate your computer to fetch a valid UrPacker license, copy and send to us before you want to purchase and install a license.

```
current host hwid: mac-arm64-01ac2ad20eeca7b90f408d6be2275192
```


VP Assistant
Hi,dear 01ac2ad20eeca7b90f408d6be2275192:
Welcome to vpand.com maintenance tool.
ClangVMP
PhoneVMP
A64Dbg
VMPStudio
UraniumVM
UrPacker
License
ClangVMP is a LLVM/Clang based native code (C/C++/ObjC/etc.) virtualized protection compiler.
The installation without a valid license is a trial version for code virtualized compilation test,the result cannot run.
If you want to apply it to your real products,please contact us to buy a license.
Standalone
Visual Studio
Xcode
Android NDK
Linux
Installation Path: /Users/geekneo/VPAssistant/product ,0.0.0
...
Install
Uninstall
Update
Cancel
...
22:16:01 I : Start downloading list.txt...
22:16:04 I : Finished dowloading list.txt.
22:16:04 I : A64Dbg remote/local version is 1.17.0/0.0.0, please install it.
22:16:04 I : ClangVMP remote/local version is 1.1.0/0.0.0, please install it.
22:16:04 I : PhoneVMP remote/local version is 1.0.0/0.0.0, please install it.
22:16:04 I : UraniumVM remote/local version is 20240405.1/0.0.0, please install it.
22:16:04 I : UrPacker remote/local version is 1.0.0/0.0.0, please install it.
22:16:04 I : VMPStudio remote/local version is 1.10.0/0.0.0, please install it.
22:16:04 I : VP Assistant v1.1.1 (Apr 12 2024), current host hwid: mac-arm64-01ac2ad20eeca7b90f408d6be2275192 .

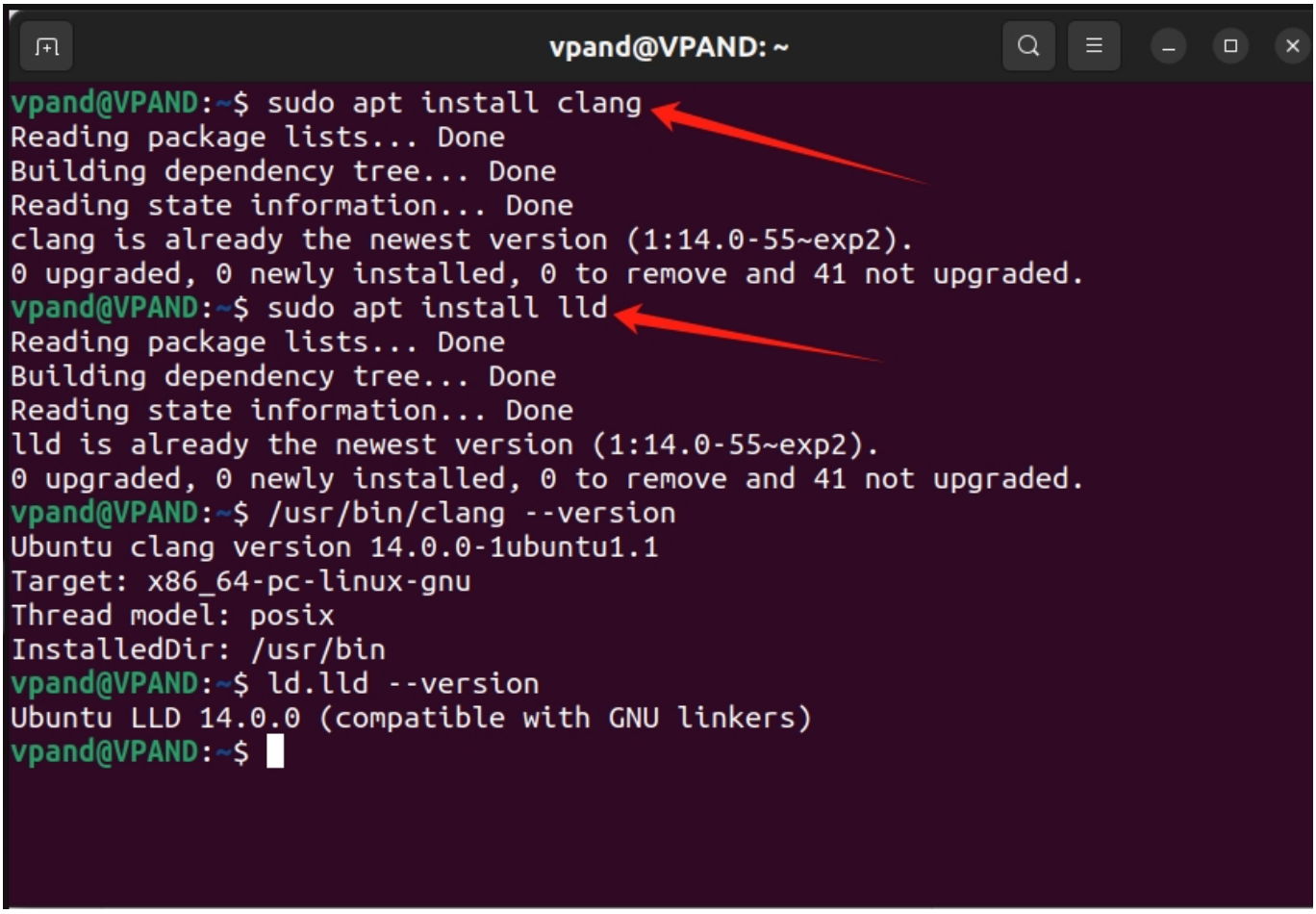

VP Assistant
Hi,dear 01ac2ad20eeca7b90f408d6be2275192:
Welcome to vpand.com maintenance tool.
ClangVMP
PhoneVMP
A64Dbg
VMPStudio
UraniumVM
UrPacker
License
A macho/elf compression shell that can make multi-binary into one,supports macho/android-elf format.
1.Dynamic linkage of liburaniumpacker.dylib/liburaniumpacker.so is free;
2.Static linkage of liburaniumpacker.a or technical support or any other custom requirements should contact us to purchase authorization;
Installation Path: /Users/geekneo/VPAssistant/product/urpacker ,1.0.0
...
Install
Uninstall
Update
Cancel
...

```
10:34:49 I : A64Dbg    remote/local version is 1.17.0/0.0.0, please install it.
10:34:49 I : ClangVMP  is the newest version 1.1.1.
10:34:49 I : PhoneVMP  is the newest version 1.0.0.
10:34:49 I : UraniumVM remote/local version is 20240405.1/0.0.0, please install it.
10:34:49 I : UrPacker  is the newest version 1.0.0.
10:34:49 I : VMPStudio is the newest version 1.10.0.
10:34:49 I : VP Assistant v1.1.3 (Apr 21 2024), current host hwid: mac-arm64-01ac2ad20eeca7b90f408d6be2275192 .
```

Linux clang/lld

Before you can apply UrPacker to target file on Linux, clang and lld should be installed as the following steps:

```
sudo apt install clang
sudo apt install lld
```



Command Line Interface

```
UrPacker@vpand.com urpacker % ./mac-arm64/upacker --help
OVERVIEW: UraniumPacker v1.0.0, A MachO/Android-ELF Compress & Pack Tool
powered by http://yunyoo.cn/ https://vpand.com/ .

USAGE: upacker [options] <input file>
```

```
OPTIONS:

Generic Options:

  --help                  - Display available options (--help-hidden for
more)
  --help-list             - Display list of available options (--help-
list-hidden for more)
  --version               - Display the version of this program

UraniumPacker Options:

  --packer-input=<string>  - UraniumPacker input macho/elf file
  --packer-libdir=<string> - UraniumPacker library search directory
  --packer-output=<string> - UraniumPacker output result file
  --packer-rtdyn           - UraniumPacker runtime type is dynamic

Pass @FILE as argument to read options from FILE.
```

- **--packer-input**: Add the binary file path to the input list. Multiple files can be specified.
- **--packer-output**: Specify the binary file output path.
- **--packer-libdir**: Specify the third-party library directory path that the input binary file depends on, which is used to provide symbolic information when the ld.lld links the final result file.
- **--packer-rtdyn**: Whether to use dynamic runtime (default). Don't forget to add liburaniumpacker.dylib or liburaniumpacker.so into your final product package, otherwise, the target process will exit because it cannot find the liburaniumpacker library;

Example

As we know, the LLVM, LLDB and Clang executable files are super big, if we want to both compress and merge the scattered files, then UrPacker is suitable. After merging, their individual symbols are still merged and remained in the final output file. If you have other modules that depend on these symbols, just need to create a soft link with the same name pointing to the result file as follows:

```
UrPacker@vpand.com lib % upacker --packer-input=libclang-cpp.dylib --
packer-input=libclang.dylib --packer-input=liblldb.dylib --packer-
input=libLLVM.dylib --packer-output=libLLVM.dylib.upacker

+> Packing libclang-cpp.dylib (1/4)...
+> Packing libclang.dylib (2/4)...
+> Packing liblldb.dylib (3/4)...
+> Packing libLLVM.dylib (4/4)...
-> Symbol _ZN4llvm3Any6TypeIdIPKNS_13LazyCallGraph3SCCEE2IdE has already
been encoded in another file.
-> Symbol _ZN4llvm3Any6TypeIdIPKNS_8FunctionEE2IdE has already been
encoded in another file.
+> libclang-cpp.dylib : 40758304 B, 39803.0 KB, 38.87lf MB
+> libclang.dylib : 22130464 B, 21611.8 KB, 21.11lf MB
+> liblldb.dylib : 15841712 B, 15470.4 KB, 15.11lf MB
+> libLLVM.dylib : 74727760 B, 72976.3 KB, 71.27lf MB
```

```
-> SUM : 153458240 B, 149861.6 KB, 146.35 MB
-> libLLVM.dylib.upacker : 59732946 B, 58333.0 KB, 56.97 MB
-> Compression rate : 38.92%
-> Success.

UrPacker@vpand.com lib % ln -s libLLVM.dylib.upacker libclang-cpp.dylib
UrPacker@vpand.com lib % ln -s libLLVM.dylib.upacker libclang.dylib
UrPacker@vpand.com lib % ln -s libLLVM.dylib.upacker liblldb.dylib
UrPacker@vpand.com lib % ln -s libLLVM.dylib.upacker libLLVM.dylib
UrPacker@vpand.com lib % ls -l

total 116672
lrwxr-xr-x  1  staff        21  5  3 10:47 libLLVM.dylib ->
libLLVM.dylib.upacker
-rwxr-xr-x  1  staff  59732946  5  3 10:47 libLLVM.dylib.upacker
lrwxr-xr-x  1  staff        21  5  3 10:47 libclang-cpp.dylib ->
libLLVM.dylib.upacker
lrwxr-xr-x  1  staff        21  5  3 10:47 libclang.dylib ->
libLLVM.dylib.upacker
lrwxr-xr-x  1  staff        21  5  3 10:47 liblldb.dylib ->
libLLVM.dylib.upacker
```

License

- **Dynamic linkage** of liburaniumpacker.dylib/liburaniumpacker.so is free.
- **Static linkage** of liburaniumpacker.a or **technical support** or any other **custom requirements** should contact us to purchase authorization, the price is 1000$/year/platform/architecture.
- **Custom compression algorithm like brotli**, the price is negotiable.

Contact us

Email

If you have any questions or problems on our products or services, feel free to contact us via email at anytime:

- **neoliu2011@gmail.com**

We-Media

Till now, we-media is our main operation method, you can also contact us via the following platforms:

- Facebook
- YouTube
- Reddit
- TikTok
- Instagram