# REmnux Usage Tips for Malware Analysis on Linux

Authored by Lenny Zeltser for [REMnux](https://REMnux.org/) v7. Lenny writes a security blog at [zeltser.com](https://zeltser.com/blog/) and is active on Twitter as [@lennyzeltser](https://twitter.com/lennyzeltser). Many REMnux tools and techniques are discussed in the   
[Reverse-Engineering Malware course](https://sans.org/for610) at SANS Institute, which Lenny co-authored. This cheat sheet is distributed according to the [Creative Commons v3 “Attribution” License](http://creativecommons.org/licenses/by/3.0/).

This cheat sheet outlines some of the commands and tools for analyzing malware using the [REMnux](https://REMnux.org/) distro.

## Get Started with REMnux

Get REMnux as a [virtual appliance](https://docs.remnux.org/install-distro/get-virtual-appliance), install the distro on a [dedicated system](https://docs.remnux.org/install-distro/install-from-scratch), or add it to an [existing one](https://docs.remnux.org/install-distro/add-to-existing-system).

Review REMnux documentation at [docs.remnux.org](https://docs.remnux.org/).

[Keep your system up to date](https://docs.remnux.org/install-distro/keep-the-distro-up-to-date) by periodically running “remnux upgrade” and “remnux update”.

Become familiar with REMnux malware analysis tools [available as Docker images](https://docs.remnux.org/run-tools-in-containers/remnux-containers).

Know default logon credentials: remnux/malware

## Operate Your REMnux System

|  |  |
| --- | --- |
| Shut down the system | shutdown |
| Reboot the system | reboot |
| Switch to a root shell | sudo -s |
| Renew DHCP lease | renew-dhcp |
| See current IP address | myip |
| Edit a text file | code *file* |
| View an image file | feh *file* |
| Start web server | httpd start |
| Start SSH server | sshd start |

## Analyze Windows Executables

Static Properties: [manalyze](https://github.com/JusticeRage/Manalyze), [peframe](https://github.com/guelfoweb/peframe), [pefile](https://github.com/erocarrera/pefile), [exiftool](https://exiftool.org/), [clamscan](https://www.clamav.net/), [pescan](http://pev.sourceforge.net/), [portex](https://github.com/katjahahn/PortEx), [bearcommander](https://github.com/hasherezade/bearparser/wiki), [pecheck](https://docs.remnux.org/discover-the-tools/examine+static+properties/pe+files#pecheck)

Strings and Deobfuscation: [pestr](http://pev.sourceforge.net/), [bbcrack](https://github.com/decalage2/balbuzard), [brxor.py](https://github.com/REMnux/distro/blob/master/files/brxor.py), [base64dump](https://blog.didierstevens.com/2020/07/03/update-base64dump-py-version-0-0-12/), [xorsearch](https://blog.didierstevens.com/programs/xorsearch/), [flarestrings](https://github.com/fireeye/stringsifter), [floss](https://github.com/fireeye/flare-floss), [cyberchef](https://github.com/gchq/CyberChef/)

Code Emulation: [binee](https://github.com/carbonblack/binee), [capa](https://github.com/fireeye/capa), [vivbin](https://github.com/vivisect/vivisect)

Disassemble/Decompile: [ghidra](https://ghidra-sre.org/), [cutter](https://cutter.re/), [objdump](https://en.wikipedia.org/wiki/Objdump), [r2](https://www.radare.org/n/radare2.html)

Unpacking: [bytehist](https://www.cert.at/downloads/software/bytehist_en.html), [de4dot](https://github.com/0xd4d/de4dot), [upx](https://upx.github.io/)

## Reverse-Engineer Linux Binaries

Static Properties: [trid](https://mark0.net/soft-trid-e.html), [exiftool](https://exiftool.org/), [pyew](https://github.com/joxeankoret/pyew), [readelf.py](https://github.com/eliben/pyelftools)

Disassemble/Decompile: [ghidra](https://ghidra-sre.org/), [cutter](https://cutter.re/), [objdump](https://en.wikipedia.org/wiki/Objdump), [r2](https://www.radare.org/n/radare2.html)

Debugging: [edb](https://github.com/eteran/edb-debugger), [gdb](https://www.sourceware.org/gdb/)

Behavior Analysis: [ltrace](https://docs.remnux.org/discover-the-tools/dynamically+reverse-engineer+code/elf+files#ltrace), [strace](https://strace.io/), [frida](https://frida.re/), [sysdig](https://github.com/draios/sysdig), [unhide](http://www.unhide-forensics.info/)

## Investigate Other Forms of Malicious Code

Android: [apktool](https://ibotpeaches.github.io/Apktool/), [droidlysis](https://github.com/cryptax/droidlysis), [androgui.py](https://github.com/androguard/androguard), [baksmali](https://bitbucket.org/JesusFreke/smali), [dex2jar](https://github.com/pxb1988/dex2jar)

Java: [cfr](https://www.benf.org/other/cfr/), [procyon](https://bitbucket.org/mstrobel/procyon), jad, [jd-gui](https://java-decompiler.github.io/), [idx\_parser.py](https://github.com/digitalsleuth/Java_IDX_Parser)

Python: [pyinstxtractor.py](https://github.com/extremecoders-re/pyinstxtractor), [pycdc](https://github.com/zrax/pycdc)

JavaScript: [js](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey), [js-file](https://blog.didierstevens.com/2018/04/19/update-patched-spidermonkey/), [objects.js](https://github.com/REMnux/salt-states/blob/master/remnux/config/objects/objects.js), [box-js](https://github.com/CapacitorSet/box-js)

Shellcode: [shellcode2exe.bat](https://github.com/repnz/shellcode2exe), [scdbg](http://sandsprite.com/blogs/index.php?uid=7&pid=152), [xorsearch](https://blog.didierstevens.com/programs/xorsearch/)

PowerShell: [pwsh](https://github.com/powershell/powershell), [base64dump](https://blog.didierstevens.com/2020/07/03/update-base64dump-py-version-0-0-12/)

Flash: [swfdump](http://swftools.org/), [flare](http://www.nowrap.de/flare.html), [flasm](http://www.nowrap.de/flasm.html), [swf\_mastah.py](https://github.com/9b/pdfxray_lite), [xxxswf](https://github.com/viper-framework/xxxswf)

## Examine Suspicious Documents

Microsoft Office Files: [vmonkey](https://www.decalage.info/en/vba_emulation), [pcodedmp](https://github.com/bontchev/pcodedmp), [olevba](http://www.decalage.info/python/oletools), [xlmdeobfuscator](https://github.com/DissectMalware/XLMMacroDeobfuscator), [oledump.py](https://blog.didierstevens.com/programs/oledump-py/), [msoffice-crypt](https://github.com/herumi/msoffice), [ssview](https://www.mitec.cz/ssv.html)

RTF Files: [rtfobj](http://www.decalage.info/python/oletools), [rtfdump](https://blog.didierstevens.com/2018/12/10/update-rtfdump-py-version-0-0-9/)

Email Messages: [emldump](https://blog.didierstevens.com/2017/07/21/update-emldump-py-version-0-0-10/), [msgconvert](https://www.matijs.net/software/msgconv/)

PDF Files: [pdfid, pdfparser](https://blog.didierstevens.com/programs/pdf-tools/), [pdfextract, pdfdecrypt](https://github.com/gdelugre/origami), [peepdf](https://eternal-todo.com/tools/peepdf-pdf-analysis-tool), [pdftk](https://gitlab.com/pdftk-java/pdftk), [pdfresurrect](https://github.com/enferex/pdfresurrect), [qpdf](http://qpdf.sourceforge.net/), [pdfobjflow](https://bitbucket.org/sebastiendamaye/pdfobjflow)

General: [base64dump](https://blog.didierstevens.com/2020/07/03/update-base64dump-py-version-0-0-12/), [tesseract](https://github.com/tesseract-ocr/tesseract), [exiftool](https://exiftool.org/)

## Explore Network Interactions

Monitoring: [burpsuite](https://portswigger.net/), [networkminer](https://www.netresec.com/), [polarproxy](https://www.netresec.com/?page=PolarProxy), [mitmproxy](https://mitmproxy.org/), [wireshark, tshark](https://www.wireshark.org/), [ngrep](https://www.tcpdump.org/), [tcpxtract](http://tcpxtract.sourceforge.net/)

Connecting: [thug](https://github.com/buffer/thug), [nc](https://nc110.sourceforge.io/), [tor](https://www.torproject.org/), [wget](https://www.gnu.org/software/wget/), [curl](https://curl.haxx.se/), [irc](http://www.epicsol.org/), [ssh](https://man.openbsd.org/ssh.1), [unfurl](https://github.com/obsidianforensics/unfurl)

Services: [fakedns](https://code.activestate.com/recipes/491264-mini-fake-dns-server/), [fakemail](https://hg.sr.ht/~olly/fakemail), [accept-all-ips](https://github.com/REMnux/distro/blob/master/files/accept-all-ips), [nc](https://nc110.sourceforge.io/), [httpd](https://nginx.org/), [inetsim](https://www.inetsim.org/), [fakenet](https://github.com/fireeye/flare-fakenet-ng), [sshd](https://man.openbsd.org/sshd.8), [myip](https://github.com/REMnux/distro/blob/master/files/myip)

## Gather and Analyze Data

Network: [Automater.py](http://www.tekdefense.com/automater/), [shodan](https://github.com/achillean/shodan-python/), [ipwhois\_cli.py](https://github.com/secynic/ipwhois), [pdnstool](https://github.com/chrislee35/passivedns-client)

Hashes: [malwoverview.py](https://github.com/digitalsleuth/malwoverview), [nsrllookup](https://github.com/rjhansen/nsrllookup), [Automater.py](http://www.tekdefense.com/automater/), [vt](https://github.com/doomedraven/VirusTotalApi), [virustotal-search.py](https://blog.didierstevens.com/programs/virustotal-tools/)

Files: [yara](https://virustotal.github.io/yara/), [scalpel](https://github.com/sleuthkit/scalpel), [bulk\_extractor](https://github.com/simsong/bulk_extractor/), [ioc\_writer](https://github.com/mandiant/ioc_writer)

Other: [dexray](http://www.hexacorn.com/blog/category/software-releases/dexray/), [viper](https://github.com/viper-framework/viper), [time-decode.py](https://github.com/digitalsleuth/time_decode)

## Other Analysis Tasks

Memory Forensics: vol.py, [vol3](https://github.com/volatilityfoundation/volatility3), [linux\_mem\_diff.py](https://github.com/monnappa22/linux_mem_diff_tool), [aeskeyfind, rsakeyfind](https://citp.princeton.edu/our-work/memory/), [bulk\_extractor](https://github.com/simsong/bulk_extractor/)

File Editing: [wxHexEditor](https://sourceforge.net/projects/wxhexeditor/), [scite](https://www.scintilla.org/SciTE.html), [code](https://code.visualstudio.com/), [xpdf](http://www.xpdfreader.com/), [convert](https://imagemagick.org/)

File Extraction: [7z](https://www.7-zip.org/), [unzip](http://infozip.sourceforge.net/), [unrar](https://www.rarlab.com/), [cabextract](https://www.cabextract.org.uk/)

## Use Docker Containers for Analysis

[Thug](https://docs.remnux.org/run-tools-in-containers/remnux-containers#thug) Honeyclient: remnux/thug

[JSDetox](https://docs.remnux.org/run-tools-in-containers/remnux-containers#jsdetox) JavaScript Analysis: remnux/jsdetox

[Rekall](https://docs.remnux.org/run-tools-in-containers/remnux-containers#rekall) Memory Forensics: remnux/recall

[RetDec](https://docs.remnux.org/run-tools-in-containers/remnux-containers#retdec) Decompiler: remnux/retdec

[Radare2](https://docs.remnux.org/run-tools-in-containers/remnux-containers#radare2) Reversing Framework: remnux/radare2

[Ciphey](https://docs.remnux.org/run-tools-in-containers/remnux-containers#ciphey) Automatic Decrypter: remnux/ciphey

[Viper](https://docs.remnux.org/run-tools-in-containers/remnux-containers#viper-binary-analysis-and-management-framework) Binary Analysis Framework: remnux/viper

[REMnux in a Container](https://docs.remnux.org/install-distro/remnux-in-a-container): remnux/remnux-distro

## Interact with Docker Images

|  |  |
| --- | --- |
| List local images | docker images |
| Update local image | docker pull *image* |
| Delete local image | docker rmi *imageid* |
| Delete unused resources | docker system prune |
| Open a shell inside a transient container | docker run --rm -it *image* bash |
| Map a local TCP port 80 to container’s port 80 | docker run --rm -it -p 80:80 *image* bash |
| Map your current directory into container | docker run --rm -it -v *.*:*dir* *image* bash |