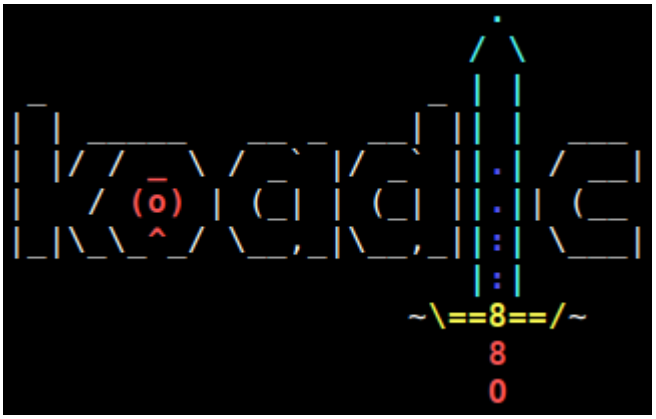
GitHub - offsecginger/koadic: zerosum0x0's Koadic

By offsecginger

Archived: 2026-04-05 16:34:51 UTC

ORIGINALLY DEVELOPED BY ZEROSUM0X0 (https://twitter.com/zerosum0x0)



Koadic, or COM Command & Control, is a Windows post-exploitation rootkit similar to other penetration testing tools such as Meterpreter and Powershell Empire. The major difference is that Koadic does most of its operations using Windows Script Host (a.k.a. JScript/VBScript), with compatibility in the core to support a default installation of Windows 2000 with no service packs (and potentially even versions of NT4) all the way through Windows 10.

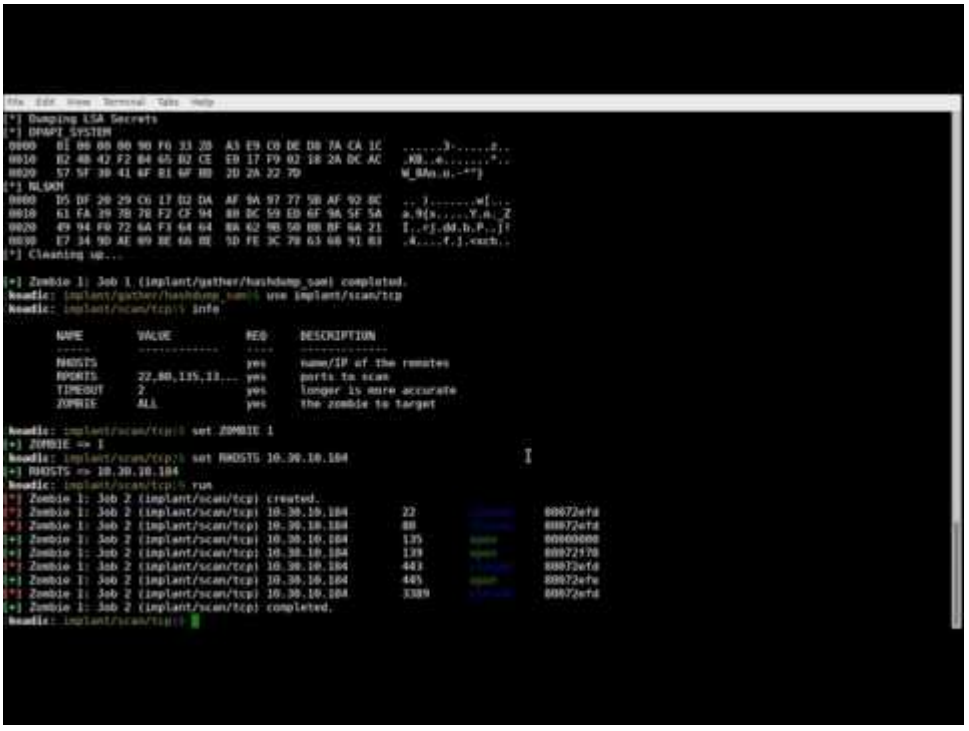
It is possible to serve payloads completely in memory from stage 0 to beyond, as well as use cryptographically secure communications over SSL and TLS (depending on what the victim OS has enabled).

Recent versions Koadic are developed on Python 3, it is not a priority to have Python 2 support (End of Life).

Install

```
 git clone https://github.com/zerosum0x0/koadic.git
cd koadic
pip3 install -r requirements.txt
./koadic
```

Demo



1. Hooks a zombie
2. Elevates integrity (UAC Bypass)
3. Dumps SAM/SECURITY hive for passwords
4. Scans local network for open SMB
5. Pivots to another machine

Stagers

Stagers hook target zombies and allow you to use implants.

Module	Description
stager/js/mshta	serves payloads in memory using MSHTA.exe HTML Applications
stager/js/regsvr	serves payloads in memory using regsvr32.exe COM+ scriptlets
stager/js/rundll32_js	serves payloads in memory using rundll32.exe
stager/js/disk	serves payloads using files on disk
stager/js/wmic	serves payloads in memory using WMIC
stager/js/bitsadmin	transfers a .wsf payload containing JScript over a Bitsadmin job and executes it

Implants

Implants start jobs on zombies.

Module	Description
implant/elevate/bypassuac_compdefaults	Bypass UAC via registry hijack for ComputerDefaults.exe.
implant/elevate/bypassuac_compmgmtlauncher	Bypass UAC via registry hijack for CompMgmtLauncher.exe.
implant/elevate/bypassuac_eventvwr	Uses enigma0x3's eventvwr.exe exploit to bypass UAC on Windows 7, 8, and 10.
implant/elevate/bypassuac_fodhelper	Bypass UAC via registry hijack for fodhelper.exe.
implant/elevate/bypassuac_sdclt	Uses enigma0x3's sdclt.exe exploit to bypass UAC on Windows 10.
implant/elevate/bypassuac_slui	Bypass UAC via registry hijack for slui.exe.
implant/elevate/system_createservice	Elevate from administrative session to SYSTEM via SC.exe.
implant/fun/zombie	Maxes volume and opens The Cranberries YouTube in a hidden window.
implant/fun/voice	Plays a message over text-to-speech.
implant/gather/clipboard	Retrieves the current content of the user clipboard.
implant/gather/comsvcs_lsass	Utilizes comsvcs.dll to create a MiniDump of LSASS, parses with pypykatz.
implant/gather/enum_domain_info	Retrieve information about the Windows domain.
implant/gather/hashdump_dc	Domain controller hashes from the NTDS.dit file.
implant/gather/hashdump_sam	Retrieves hashed passwords from the SAM hive.
implant/gather/loot_finder	Finds loot on the target box.
implant/gather/user_hunter	Locate users logged on to domain computers (using Dynamic Wrapper X).
implant/inject/mimikatz_dotnet2js	Injects a reflective-loaded DLL to run powerkatz.dll (@tirannido DotNetToJS).
implant/inject/mimikatz_dynwrapx	Injects a reflective-loaded DLL to run powerkatz.dll (using Dynamic Wrapper X).
implant/inject/mimikatz_tashlib	Executes arbitrary shellcode using the TashLib COM object. (Work in Progress!)

Module	Description
implant/inject/shellcode_dotnet2js	Executes arbitrary shellcode using the DotNet2JS technique. Inject shellcode into a host process via createremotethread as a new thread (thanks psmitty7373!).
implant/inject/shellcode_dynwrapx	Executes arbitrary shellcode using the Dynamic Wrapper X COM object.
implant/inject/shellcode_excel	Runs arbitrary shellcode payload (if Excel is installed).
implant/manage/enable_rdesktop	Enables remote desktop on the target.
implant/manage/exec_cmd	Run an arbitrary command on the target, and optionally receive the output.
implant/persist/add_user	Adds a either a local or domain user.
implant/persist/registry	Adds a Koadic stager payload in the registry.
implant/persist/schtasks	Establishes persistence via a scheduled task.
implant/persist/wmi	Creates persistence using a WMI subscription.
implant/phishing/password_box	Prompt a user to enter their password.
implant/pivot/exec_psexec	Run a command on another machine using psexec from sysinternals.
implant/pivot/exec_wmi	Executes a command on another system.
implant/pivot/stage_wmi	Hook a zombie on another machine using WMI.
implant/scan/tcp	Uses HTTP to scan open TCP ports on the target zombie LAN.
implant/utils/download_file	Downloads a file from the target zombie.
implant/utils/multi_module	Run a number of implants in succession.
implant/utils/upload_file	Uploads a file from the listening server to the target zombies.

Future Improvements (a.k.a. Koadic 2.0)

- Rewrite and redesign the server stack to be cleaner.
- Actual VBScript support.

Disclaimer

Code samples are provided for educational purposes. Adequate defenses can only be built by researching attack techniques available to malicious actors. Using this code against target systems without prior permission is illegal in most jurisdictions. The authors are not liable for any damages from misuse of this information or code.

Creators

- @Aleph___Naught
- @The_Naterz
- @JennaMagius
- @zerosum0x0

Contributors

- @vvalien1
- fbctf
- cclaus
- Arno0x
- delirious-lettuce
- psmitty7373
- @ForensicITGuy

Acknowledgements

Special thanks to research done by the following individuals:

- @subTee
- @enigma0x3
- @tiraniddo
- @harmj0y
- @gentilkiwi
- @mattifestation
- clymb3r
