CQURE

CQTools:

The New Ultimate Hacking Toolkit

by

Paula Januszkiewicz & CQURE Team

(paula@cqure.us)

Warsaw New York Dubai Zug

1 Abstract

CQURE Team has prepared tools used during penetration testing and packed them in a toolkit called CQTools. This toolkit allows to deliver complete attacks within the infrastructure, starting with sniffing and spoofing activities, going through information extraction, password extraction, custom shell generation, custom payload generation, hiding code from antivirus solutions, various keyloggers and leverage this information to deliver attacks. Some of the tools are based on discoveries that were released to the world for the first time by CQURE Team. CQURE was the first team that did a full reverse engineering of DPAPI (Data Protection Application Programming Interface) and prepared the first public tool that allows to monitor WSL (Windows Subsystem for Linux) feature.

2 CQTools technical details

A detailed description of tools in CQTools toolkit is provided below.

CQWSLMon.exe

Windows Subsystem for Linux (WSL) is a compatibility layer for running Linux binary executables (in ELF format) natively on Windows 10 and Windows Server 2019. CQWSLMon is the first publicly know tool that allows to monitor the interaction with the subsystem.

CQRegKeyLastWriteTime.exe

Allows to extract information about the datetime when the Registry Key was modified for the last time. This information may be helpful in forensics or malicious code development (to know what trails are generated by the code).

```
Usage: CQRegKeyLastWriteTime.exe <reg_key>
```

CQNTDSDTDecrypter.exe

Decrypts ntds.dit file by providing appropriate Bootkey, extracts password hashes, KDS master root keys. More details: https://cqureacademy.com/blog/windows-internals/data-protection-api

```
Usage: CQNTDSDTDecrypter /bootkey /file
Available parameters:
      --bootkey=VALUE        The bootkey, extracted from the registry.
      --file, --ntds=VALUE   The ntds.dit file containing the AD data.
      --outfile, --out=VALUE The text file containing decrypted password
                               hashes.
      --pfxfile, --pfx=VALUE The file containing dpapi pfx.
      --kdsrootkeyfile, --kds, --kdsrootkey=VALUE
                             The file containing dpapi-ng Group Key
                               Distribution Service master root key.
```

CQLsassSecretsDumper.exe

Dumps DPAPI Golden Key (Backup key) from LSASS to pfx file. When DPAPI is used in Active Directory domain environment, a copy of user's master key is encrypted with a so-called DPAPI Backup Key. Windows Server 2000 uses a symmetric key and newer systems use a public/private key pair. If the user password is reset and the original master key is rendered inaccessible to the user, the user's access to the master key is automatically restored using the backup key. DPAPI Backup Key cannot be changed, so the leakage of the key may result in the need for reconfiguration of the whole environment.

```
Usage: CQLsassSecretsDumper /file
Available parameters:
  -h, -?, --help             This help
      --file, -f=VALUE       The output file name
```

CQDPAPIExportPFXFromAD_mimikatz_way.exe

Extracts DPAPI Golden Key in pfx format from AD the same way Mimikatz does

```
Usage: CQDPAPIExportPFXFromAD_mimikatz_way /file
Available parameters:
  -h, -?, --help             This help
      --file, -f=VALUE       The output file name
```

CQMasterKeyAD.exe

Allows decryption of DPAPI protected data by leveraging usage of the private key stored as LSA Secret on a domain controller (we have called it a 'backup key,' and this is a key corresponding to the backup public key stored in the domain user's profile). The backup key allows decrypting literally all of the domain user's secrets (passwords / private keys/information stored by the browser). In other words, someone who has the backup key is able to take over all of the identities and their secrets within the whole enterprise.

```
Usage: CQMasterKeyAD /file /pfx /newhash
Available parameters:
      --pfx=VALUE            Path to the pfx file containing RSA private key
                               (DPAPI Golden Key).
      --file=VALUE           Path to the Masterkey file.
      --newhash=VALUE        MD4 or SHA1 (but the same algo as for oldhash!)
                               for new masterkey. In AD environment and domain
                               accounts most probably MD4, in standalone: SHA1.
```

CQDPAPIBlobDecrypter.exe

Decrypts Blob with DPAPI. This tool has unique feature of using masterkey for decryption instead of WINAPI and providing password like most of the decrypters.

```
Usage: CQDPAPIBlobDecrypter /masterkey /goldenkeyfile
Available parameters:
      --master=VALUE         The masterkey provided as a hex string.
      --entropy=VALUE        Entropy used during encryption.
      --blob, --blobfile=VALUE
                             The binary file containing blob itself
      --out, --outfile=VALUE Text file containing decrypted blob in hextext
```

CQDPAPIBlobSearcher.exe

Search for DPAPI blobs inside a file.

```
Usage: CQDPAPIBlobSearcher /file /outdir

Available optional parameters:
  -f, --file=VALUE           File to be searched
  -d, --dir=VALUE            Directory to be searched
      --reg, --regkey=VALUE  Registry key to be searched
  -r                         Search recursively
  -o, --outdir=VALUE         Path to a directory to store the DPAPI blobs
                               extracted from the file
```

CQDPAPIEncDec.exe

Encrypts and decrypts text using DPAPI.

CQDPAPIKeePassDBDecryptor.exe

Allows to decrypt KeePass database by using DPAPI data that is possessed from the domain. It provides access to all users' KeePass databases and it uses DPAPI data leveraged by CQMasterKeyAD. The tool uses decrypted Master Key of the user in order to decrypt key that encrypts KeePass database.

The tool will try to save reencrypted file to the same directory, as the original. The password used to reencrypt is 'cqure' without quotes.

```
Usage: CQDPAPIKeePassDBDecryptor /key /file
Available parameters:
  -k, --key=VALUE            The key decrypted from the DPAPI blob.
  -f, --file=VALUE           The KeePass database file.
```

CQDPAPINGPFXDecrypter.exe

Leverages DPAPI-NG used in the SID-protected PFX files. The tool allows to decrypt SID-protected PFX files even without access to user's password but just by generating the SID and user's token. More details: https://cqureacademy.com/blog/windows-internals/data-protection-api

```
Usage: CQDPAPINGPFXDecrypter /pfx /master
Available parameters:
      --pfx=VALUE             The pfx file exported with sid-based security.
      --masterkey, --master, -m=VALUE
                              The hex string containing msKds-RootKeyData
                                attrib data.
```

CQRDCManDecrypter.exe

Decrypts RDCMan .rdg files with provided masterkey and extracts credentials from it.

```
Usage: CQRDCManDecrypter /file /master
Available parameters:
      --file, -f=VALUE        Path to the .rdg file of the Remote Desktop
                                Connection Manager.
      --master, -m=VALUE      The Masterkey decrypted. You can specify more
                                than one masterkey, simply add another /master
```

CQMasterKeyDecrypt.exe

Decrypts service masterkey from MS SQL Server that is protected by DPAPI. It may be used to bypass TDE (Transparent Data Encryption) protection. It's the only publicly known tool for that purpose on the market.

```
Usage: CQMasterKeyDecrypt /masterkey /goldenkeyfile
Available parameters:
      --sid=VALUE             The sid of the user.
      --hash=VALUE            The pwdhash calculated from user password.
      --golden=VALUE          The file with golden key. You don't have to
                                specify the hash and the sid.
      --file, --masterkeyfile=VALUE
                              The masterkey file to be decrypted.
```

CQMasterKeyEncrypt.exe

Encrypts masterkey with a new hash.

```
Usage: CQMasterKeyEncrypt /sid /file /oldhash /newhash
Available parameters:
      --sid=VALUE            SID of the masterkey owner.
      --file=VALUE           Path to the Masterkey file.
      --oldhash=VALUE        MD4 or SHA1 hash used to encrypt the masterkey.
      --newhash=VALUE        MD4 or SHA1 (but the same algo as for oldhash!)
                               for new masterkey. In AD environment and domain
                               accounts most probably MD4, in standalone: SHA1.
```

CQETWKeylogger.exe

Keylogger based on ETW (Event Tracing for Windows). It only uses features built in Windows system, so no additional software is needed to perform the attack.

CQCreateProcessWithParent.exe

Allows to choose a process that will be a parent for the executed process. It enables the attacker to hide original parent process from Sysmon and makes the forensic investigation much more difficult.

```
Usage: CQCreateProcessWithParent /ppid /exe
Available parameters:
      --ppid=VALUE           The PID of the process to become a parent.
      --exe=VALUE            Exe to launch.
```

CQDGAGenerator.exe

Generator of domain names, based on the Domain Generation Algorithm - https://en.wikipedia.org/wiki/Domain_generation_algorithm , e.g. used by CryptoLocker. The generator may be used by the attacker to hide command and control server.

```
Usage: CQDGAGenerator [/from /to]
If run without params, produces 30 addresses, starting from today.

Available optional parameters:
      --fmt={0               Domain display format, eg: {0}.com
      --from=VALUE           Starting date, in format: yyyy-mm-dd. If 'to'
                               param is omitted, 30 addresses are calculated,
                               starting from 'from' date.
      --to=VALUE             End date, in format: yyyy-mm-dd. Requires 'from'
                               param.
```

CQElevate

Exploits MS16-032 vulnerability. The bug relies on how handles are processed in multiprocessor systems including Windows 10 and Windows Server 2012 R2. It relies heavily on FuzzySecurity code published in GitHub. Details: https://cqureacademy.com/blog/malware/elevation-from-regular-user-to-the-localsystem-notes-from-microsoft-ignite-2016

```
Usage: Elevate-Cmd <command>
```

CQImpersonate.exe

This tool allows to run a command in the context of any of the authenticated users from your system. This tool requires to be run in the LOCAL_SYSTEM context.

```
Usage: CQImpersonate /exe /user
Available parameters:
  -u, --user=VALUE           the username for the token
  -c, --cmd=VALUE            exe name to be run.
```

CQFindBin.exe

Searches for patterns in files.

```
CQFindBin <pattern> <file|dir>
```

CQHashesCalc.exe

MSDCC2 and NTHash calculator.

CQHashDumpv2.exe

Allows to dump hashes from the system and change passwords of the users. It's one of the few tools on the market that allows to do it both offline and online.

```
Usage: CQHashDumpv2 /samdump /dccdump /sam /sec /sys
Available parameters:
      --samdump              Dump hashes from the SAM database
      --dccdump              Dump Domain Cached Credentials
      --sam=VALUE            Path to the SAM reg file
      --sec=VALUE            Path to the SECURITY reg file
      --sys=VALUE            Path to the SYSTEM reg file
      --newmsdcc=VALUE       Binary string with new MSDCC2
      --pass=VALUE           New password
      --user=VALUE           User name for new MSDCC2
Providing any: /sam /sec or /sys switch enables offline analysis.
In offline mode /samdump enforces /sam and /sys, and /dccdump enforces /sys and
/sec.
Online mode requires access to the SECURITY registry, which by default is
accessible only by the SYSTEM account.
```

CQSecretsDumper.exe

Allows to dump credentials. Details: https://cqureacademy.com/blog/secure-server/how-to-use-group-managed-service-accounts-gmsa-vs-service-accounts.

```
CQSecretsDumper /secret /service /sec /sys
Available parameters:
      --verbose              Enable full data output (before interpretation
                               of first 16 bytes)
      --bootkey              Dump bootkey from the SYSTEM hive
      --service=VALUE        Dump password data for the service
      --secret=VALUE         Dump decrypted data from the secret
      --sec=VALUE            Path to the SECURITY reg file
      --sys=VALUE            Path to the SYSTEM reg file
Providing any: /sec or /sys switch enables offline analysis.
In offline mode you have to provide both: /sys and /sec files
Online mode requires access to the SECURITY registry, which by default is
accessible only by the SYSTEM account.
```

CQmimi64.exe

CQURE Edition of Mimikatz with additional modules.

CQMSGDecode.exe

Decodes MSG files.

```
Usage: CQMSGDecode <email.msg>
```

CQPfxRegenerator.exe

Regenerates PFX files.

```
Usage: PfxRegenerator /inkey /out /in [/in]
Available parameters:
      --in=VALUE             Path to the cert file (.cer). Can be reused to
                               create certs chain
      --inkey=VALUE          Path to the RSA key file (.rsaxml.txt)
      --out=VALUE            Path to the output pfx file
```

CQPrefetchParser.exe

This tool allows you to inspect prefetch files. Additionally, you can decompress the file (Windows 10 and newer only) and analyze it manually.

```
Usage: CQPrefetchParser /file /dir /out
Available parameters:
      --analyze, -a          Analyze the file
      --decompres, -d        Decompress the file
      --dir=VALUE            Path to the directory containing prefetch files
      --file, -f=VALUE       Path to the .pf file
      --out, -o=VALUE        Path to the decompressed .pf file (or directory,
                               where the files are going to be stored, if you
                               choose the /dir option)
```

CQEVTXRecovery.exe

Tries to repair corrupted eventlog files from [in] directory and place repaired into the [out] directory.

```
Usage: EVTXRecovery -in -out:
Available parameters:
      --in, --indir=VALUE    directory path containing corrupted eventlog
                               files
      --out, --outdir=VALUE  directory path to store repaired eventlog files
      --file, --infile=VALUE corrupted eventlog file
```

CQRdcache.exe

Allows to extract the images of the desktop from RDP cache.

CQReflectivePELoader.exe

Reflective PE packer.

```
Usage: CQReflectivePELoader exefile
```

CQRegTool.exe

Registry analyzer.

```
Usage: CQRegTool /path /file
Available parameters:
      --path=VALUE           Path to the key containing the class
      --file=VALUE           Path to reg file (offline mode)
In offline mode you have to provide both: /path and /file
```

CQARPSpoofer.exe

Allows to perform ARP spoofing attack.

```
Usage: CQArpSpoof /clientip /gwip
Available parameters:
      --clientIP, --client=VALUE
                             The ip address of the client.
      --gwIP, --gw=VALUE     The ip address of the gateway, server ip.
```

CQCat.exe

Modified netcat, networking utility for reading from and writing to network connections, that enables the attacker to bypass most of AV systems.

```
[v1.11 NT www.vulnwatch.org/netcat/]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [options] [hostname] [port]
options:
        -d              detach from console, background mode

        -e prog         inbound program to exec [dangerous!!]
        -g gateway      source-routing hop point[s], up to 8
        -G num          source-routing pointer: 4, 8, 12, ...
        -h              this cruft
        -i secs         delay interval for lines sent, ports scanned
        -l              listen mode, for inbound connects
        -L              listen harder, re-listen on socket close
        -n              numeric-only IP addresses, no DNS
        -o file         hex dump of traffic
        -p port         local port number
        -r              randomize local and remote ports
        -s addr         local source address
        -t              answer TELNET negotiation
        -u              UDP mode
        -v              verbose [use twice to be more verbose]
        -w secs         timeout for connects and final net reads
        -z              zero-I/O mode [used for scanning]
port numbers can be individual or ranges: m-n [inclusive]
```

CQReverseShellGen.exe

Generates TCP reverse shell exe file.

```
Usage: CQReverseShellGen /ip /port
Available parameters:
      --ip=VALUE             IP Address or hostname
      --port=VALUE           Port number
```

CQRunInAppContainer.exe

Runs application in AppContainer.

```
Usage: CQRunInAppContainer /exe /app
Available parameters:
      --exe=VALUE            Path to the exe to be launched in AppContainer
      --app=VALUE            AppContainer name. If not set, default:
                               CQAppContainer
```

CQSymbolInstaller.exe

Symbol installer.

```
Usage: CQSymbolInstaller /image /pdb /symstore
Available parameters:
      --image=VALUE          Path to the executable (.exe, .dll, .sys),
                               containing debug info (in RSDS format).
      --pdb=VALUE            Path to the symbol file.
      --symstore=VALUE       Path to the symstore directory.
```

CQTools license: Freeware.

3 Conclusion

CQTools provide not only features that could be used for exploitation, but also, they provide information that could be useful for security researchers such as information extracted from DPAPI or WSL (Windows Subsystem for Linux) and other information regarding Windows internals. CQTools is a useful toolkit for both delivering a penetration test and security research.

4 About the Author

Paula Januszkiewicz is an IT Security Auditor and Penetration Tester, Microsoft Regional Director, Enterprise Security MVP and trainer (MCT), and Microsoft Security Trusted Advisor. She is also a top speaker at many well-known conferences including TechEd North America, TechEd Europe, TechEd Middle East, RSA, TechDays, and CyberCrime, often rated as a number-one speaker. She has engaged as a keynote speaker for security-related events and writes articles on Windows Security. She drives her own company, CQURE, working on security-related issues and projects. She has conducted hundreds of IT security audits and penetration tests, some for governmental organizations. Her distinct specialization is on Microsoft security solutions – she holds multiple Microsoft certifications, and is familiar with and possesses certifications in other related technologies. In private, she enjoys researching new technologies, which she converts to authored trainings. She wrote a book about Threat Management Gateway 2010 and is now working on her next book. She was granted access to Windows source code.

5 About CQURE

CQURE is a provider of specialized services in IT infrastructure security, business applications, consulting and advisory services. CQURE provides the following services: high quality penetration tests with useful reports, configuration reviews, incident response emergency services, security architecture and design advisory, forensics investigation, security trainings and awareness for management and employees.

Website: https://www.cqure.pl/

E-mail: info@cqure.pl

CQURE Headquarters – Aleje Jerozolimskie 134, 02-305 Warsaw, Poland
CQURE Western Europe – Bahnhofstrasse 21 6300 Zug, Switzerland
CQURE North America – 1250 Broadway 36th Flr. New York, NY 10001, USA
CQURE Middle East – Eomarat Atrium 423, Al Wasl Area 112344 Dubai, UAE

6 Notable presentations

Contact us:
info@cqure.pl

Warsaw New York Dubai Zug