New version of chinoxy backdoor using COVID19 alerts document lure

medium.com/@Sebdraven/new-version-of-chinoxy-backdoor-using-covid19-document-lure-83fa294c0746

Sebdraven March 20, 2020

Sebdraven

Mar 20, 2020

.

4 min read

Last year I've analyzed a chinoxy backdoor dropped by an royal road RTF targeting Vietnam. https://medium.com/@Sebdraven/winnti-uses-the-rtf-exploit-8-t-too-targets-vietnam-13300d432272

The 17 march 2019, a campaign using royal road RTF targetted the Kirghistan with a lure document COVID19 about financial consideration of the world Bank.

1527f7b9bdea7752f72ffcd8b0a97e9f05092fed2cb9909a463e5775e12bd2d6 (MD5…

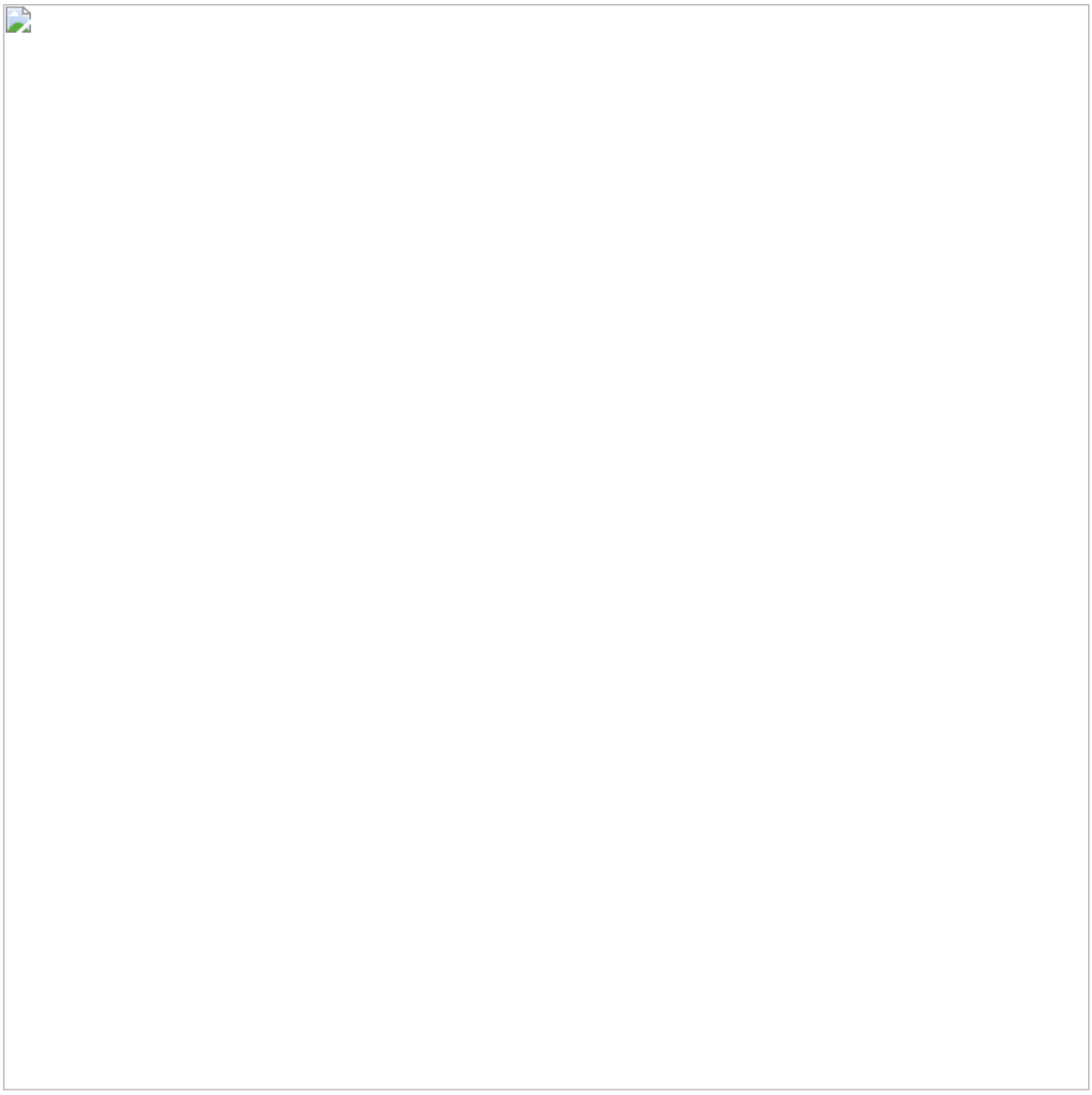
Interactive malware hunting service. Any environments ready for live testing most type of threats. Without install…

app.any.run

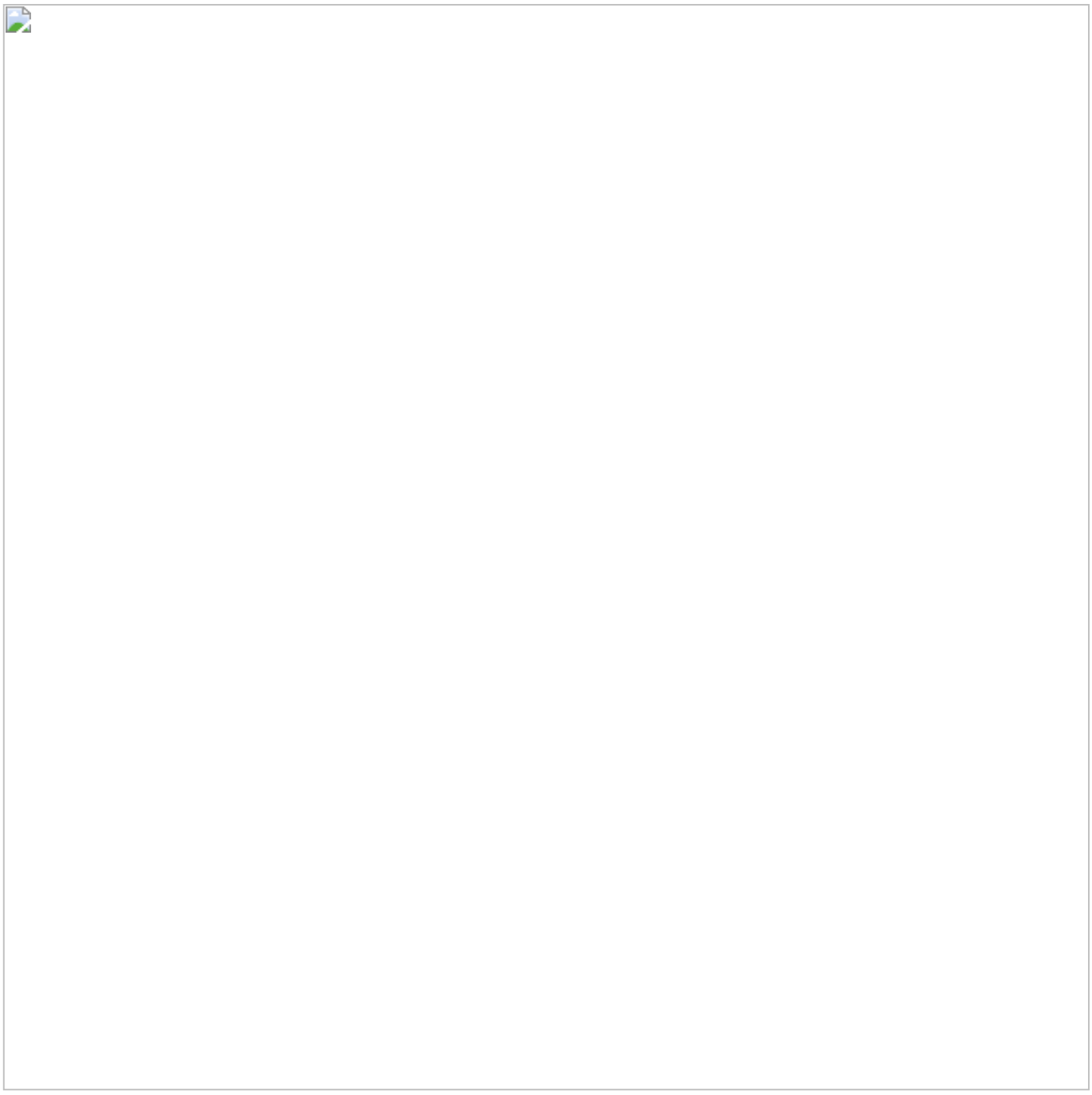
5 years of Chinoxy implemention

This backdoor is very similar with it used for the Vietnam.

We have the same protocole HTTPs custom.

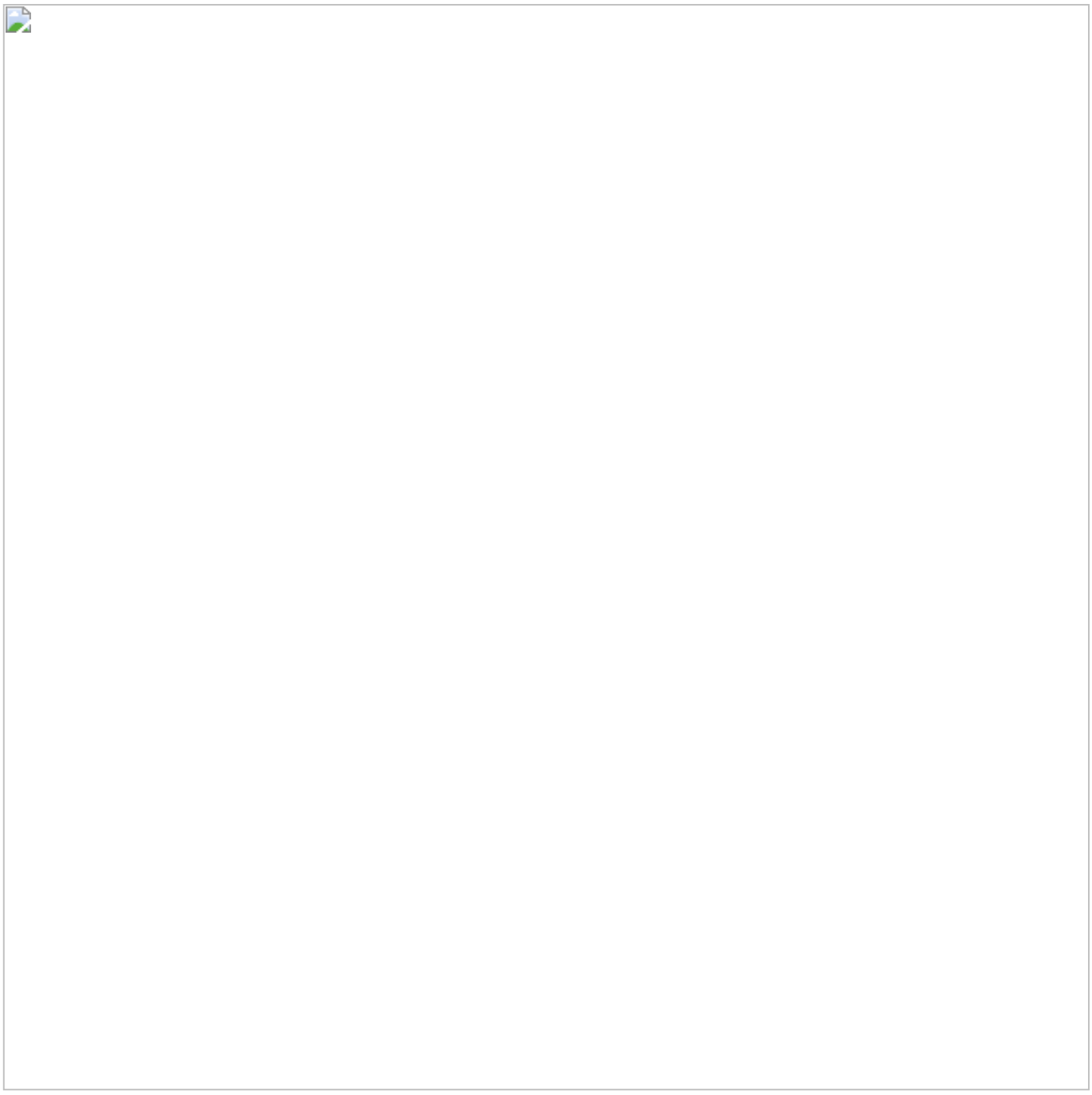


The state machine of the backoors are similare.



But in the new version, we found a functionality coming for the 2014 version.

A keylogger is implemented



The big difference with another versions, the configuration of the backdoor is in the resource named “NNKK”.

The malware reads the resource (Fun_10005c50) and decode the configuration. (FUN_10005bf0)

*pHVar1 =FindResourceW(param_2,u_NNKK_100159fc,u_TYPELIB_1001a290); if (pHVar1 != (HRSRC)0x0) { hModule = LoadLibraryW(u_kernel32.dll_10019720); local_b = 0x65; local_5 = 0x65; local_10 = ‘L’; local_f = 0x6f; local_e = 0x61; local_d = 100; local_c = 0x52; local_a = 0x73; local_9 = 0x6f; local_8 = 0x75; local_7 = 0x72; local_6 = 99; local_4 = (undefined4 *) ((uint)local_4 & 0xffffff00); pFVar2 = GetProcAddress(hModule,&local_10); hResData = (HGLOBAL)(*pFVar2)(param_2,pHVar1); FreeLibrary(hModule); if (hResData !=*

*(HGLOBAL)0x0) { puVar4 = (undefined4 *)LockResource(hResData); iVar3 = 0x4c0; puVar5 = local_4; while (iVar3 != 0) { iVar3 = iVar3 + -1; *puVar5 = *puVar4; puVar4 = puVar4 + 1; puVar5 = puVar5 + 1; } decode_resource((int)local_4,0x1300);*

In decode_resource the algorithm is very simple:

In the DATA section there are two keys to decode the configuration:*“* 22135987565*”* and “3*6969856569”*

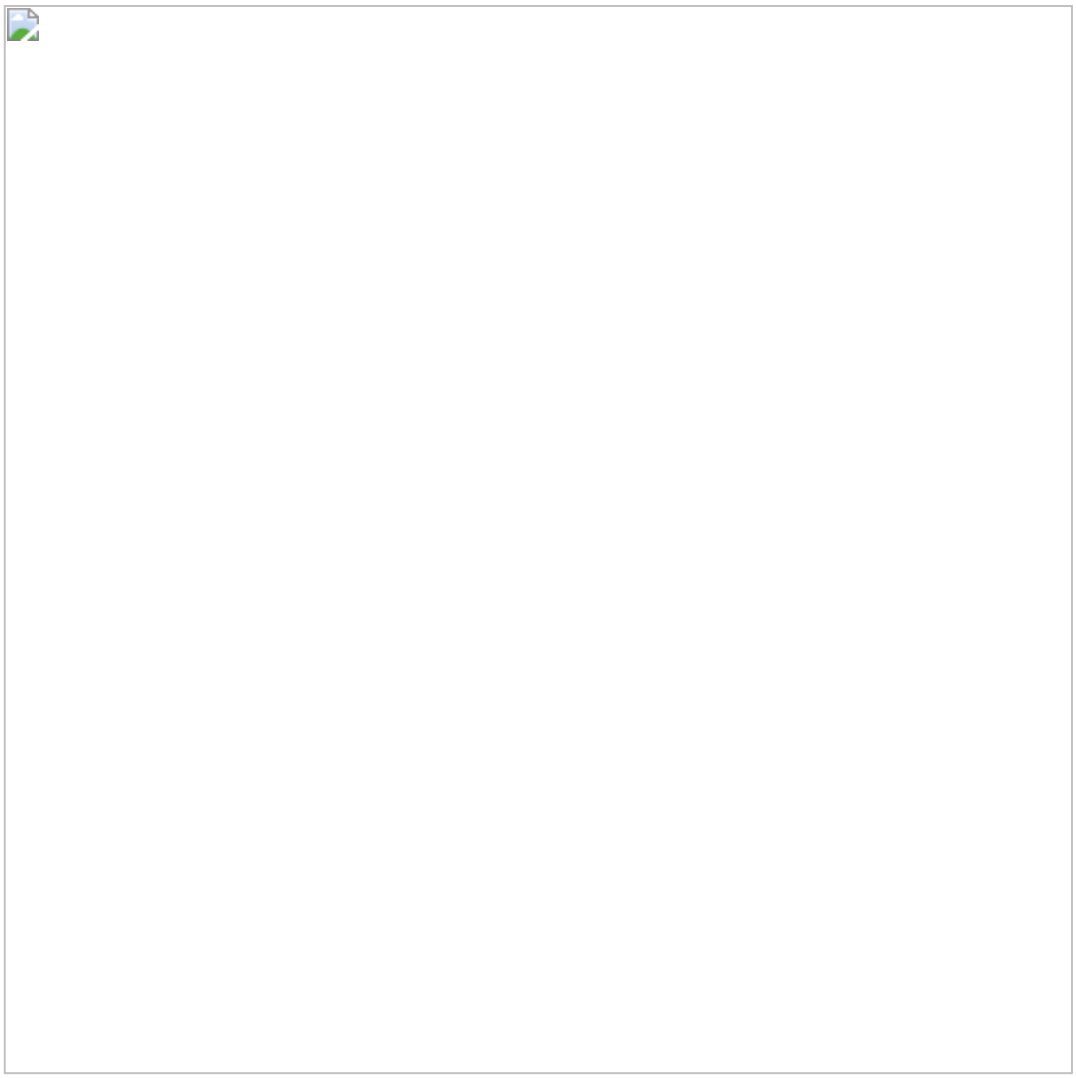
DAT_1001a278 XREF[1]: decode_ressource:10005c14(R) 1001a278 32 undefined1 32h s_2135987565_1001a279 XREF[1]: decode_ressource:10005c14(R) 1001a279 32 31 33 ds “2135987565” 35 39 38 37 35 36

DAT_1001a284 XREF[1]: decode_ressource:10005c0e(R) 1001a284 33 undefined1 33h s_6969856569_1001a285 XREF[1]: decode_ressource:10005c0e(R) 1001a285 36 39 36 ds “6969856569” 39 38 35 36 35 36

Each bytes of this keys are xored, a mask is applied with & 0x27 and there is a new xor with *DAT_1001a284*

This result is xored with each byte of the resource NNK.

the result is the config of the c2 in base64



The backdoor check it if the configuration is not stored in the registry in function FUN_10010e10.

In fact the first version of chinoxy stored the configuration in the registration at the installation:

> *add_key((HKEY)&DAT_0042d708,&DStack12,*
> *(LPDWORD)0x0,u_SYSTEM\CurrentControlSet\Service_00416214,*
> *u_Group_004170c8); DStack12 = 0x50; add_key((HKEY)&DAT_0042d758,&DStack12,*
> *(LPDWORD)0x0,u_SYSTEM\CurrentControlSet\Service_00416214,*
> *u_Remark_004161e8); DStack12 = 0x50; add_key((HKEY)&DAT_0042d6b8,&DStack12,*
> *(LPDWORD)0x0,u_SYSTEM\CurrentControlSet\Service_00416214,*
> *u_PassWord_00417158); DStack12 = 0x80;*
> *add_key((HKEY)&DAT_0042d7a8,&DStack12,*
> *(LPDWORD)0x0,u_SYSTEM\CurrentControlSet\Service_00416214,*
> *u_Version_00417148);*

We found the same the same keywords of the backdoor of vietnam in k.ini file

Group,Remark,Version,UID.

and in the new version:

u_Group_1001a754 XREF[2]: 1001135e(*), 1001214b(*) 1001a754 47 00 72 unicode u"Group" 00 6f 00 75 00 70 u_Remark_1001a760 XREF[2]: 100114ae(*), 100120e5(*) 1001a760 52 00 65 unicode u"Remark" 00 6d 00 61 00 72 1001a76e 00 ?? 00h 1001a76f 00 ?? 00h u_System_1001a770 XREF[1]: 1001152c(*) 1001a770 53 00 79 unicode u"System" 00 73 00 74 00 65 1001a77e 00 ?? 00h 1001a77f 00 ?? 00h

The config in bas64 of the backdoor is decoded in FUN_100074c0.

This function calls many time FUN_100074a0 to decode the string encoded.

int __cdecl FUN_100074a0(uint param_1)

{
int iVar1;

iVar1 = 0; do { if ((param_1 & 0xffff) == (int)s_ABCDEFGHIJKLMNOPQRSTUVWXYZabcdef_1001a440[iVar1]) { return iVar1; } iVar1 = iVar1 + 1; } while (iVar1 < 0x40); return 0;}

The result of this decoding function is:

018DDA4C 018DF780
L"brands.newst.dnsabr.com:3010|brands.newst.dnsabr.com:3010|ru.mst.dns-cloud.net:3010|"

so two domains.

The configuration is encoded, and split with '|' with the same technics of the backdoor used in Vietnam.

And to communicate with the C2, the malware uses raw socket and ws32_dll.

Threat Intelligence Consideration

Royal Road is massively used by Chinese Threat Actors. The backdoor chinoxy is used by too by this group.

So with a good confidence, this attack is driven by China againt Kirghistan.

IOCs:

Royal Road RTF:

1527f7b9bdea7752f72ffcd8b0a97e9f05092fed2cb9909a463e5775e12bd2d6

backdoor: 30115717d20e469e7c4bf45489f6c6d8810f32b1b68b6aa4b0ffcb21764ea99c

backdoor 2014:

46876d952e152573069fa15b70caf825e4bf97ffb90c00f80d26890a9d92f05b

Domains:

brands.newst.dnsabr.com:3010

ru.mst.dns-cloud.net:3010

IPs:

45.76.218.232