An Inside Look at How Ryuk Evolved Its Encryption and Evasion Techniques

(I) labs.sentinelone.com/an-inside-look-at-how-ryuk-evolved-its-encryption-and-evasion-techniques/

Marco Figueroa



Introduction

In the last three months, there has been a <u>50% uptick</u> in ransomware, with the Ryuk ransomware garnering the most attention after a string of high profile attacks that have been crippling companies. Last month it was <u>reported</u> that Ryuk hit UHS hospital networks with force, spreading across UHS healthcare facilities in the US from coast to coast. This well-orchestrated attack left many hospital workers without access to labs, radiology, and patient records, which led to workers having to resort to pen and paper to triage patients. Ryuk is currently attacking approximately 20 organizations a week, and this number will only expand due to its successes.

There are a number of factors that contribute to this success. These include its harnessing of other "toolkits" such as TrickBot and Emotet, and being quick to jump on newly-exposed vulnerabilities such as Zerologon. We also see that Ryuk has iterated since its earlier incarnations to evade detection and markedly improve the time it takes from execution to full encryption, making life increasingly difficult for organizations that cannot respond to threats at such speed.

In this post, we look at how Ryuk has evolved since 2018 and explore the improvements in encryption speed and evasion techniques that we see in Ryuk samples today. Along the way, we detail a method analysts can use to extract the Ryuk executable from memory and dump it to file for further inspection.

Ryuk Overview

When <u>Ryuk ransomware</u> burst onto the scene, it was initially believed that it was developed by the same threat actors who developed Hermes Ransomware. However, it was later discovered that Hermes was being sold on the black market, allowing cybercriminals to purchase the framework and convert it to what is known today as Ryuk.

The current waves of attacks have been known to use a combination of <u>Emotet</u>, <u>Trickbot</u>, and Ryuk. In recent weeks, the actors behind Ryuk have even been observed using <u>ZeroLogon</u> to extend their reach and broaden the delivery of their ransomware payloads. While the Ryuk payloads do not specifically contain the ZeroLogon functionality, the flaw is being leveraged at earlier stages in the attack chain. Attackers are able to use existing capabilities in <u>Cobalt Strike</u> and similar frameworks to achieve the privilege escalation. It is quickly becoming clear that ZeroLogon will become a staple in the attackers' collective "toolbelt".

Reversing and Comparing Ryuk 2018 and 2020

There are many tools in the reversing ecosystem for diffing binaries like <u>Bindiff</u>, but a fast tool when comparing binaries I find most useful is Ghidra's version tracking to check for comparisons between binary files.

If we compare an earlier version of Ryuk with the latest version, we can note some interesting changes. In the most recent version, Ryuk obfuscates its hardcoded strings to become more difficult for AV vendors to detect:



Figure 1: Ryuk 2018 vs 2020

Ryuk 2020 also copies itself to increase the speed of encryption, which we discuss in detail below.

The ransomware uses RSA and AES to encrypt files with extension **. r**y**k**, creating a new thread for each file it encrypts. Ryuk also uses the **CryptGenRandom** API, which fills the buffer with random bytes to generate a data encryption key.

		748270	95 803538468374	mov	esi,dword ptr [bcrypt:BCryptResolveProviders+0x54b8 (74834638)]
74823d30	bcrypt!BCryptOpenAlgorithmProvider (<no info="" parameter="">)</no>	74827b	9b 8bce		ecx.esi
74824270	bcrypt!BCryptCloseAlgorithmProvider (<no info="" parameter="">)</no>	74827b	9d ff1568518374	call	dword ptr [bcrvpt!BCrvptResolveProviders+0x5fe8 (74835168)]
748242f0	bcrypt BCryptEnumAlgorithms (<no info="" parameter="">)</no>	74827h	a3 ffd6	call	
748246d0	bcryptlBCryptEnumProviders (<no info="" parameter="">)</no>	74827h	a5 8hc7	mov	eav edi
74824a30	hcryotlBCcyptEreeBuffer (<pre>son parameter info>)</pre>	74827h	a7 8d65e0	lea	esn [ehn-20h]
74824580	hcryntiB(cyntGetProperty ((no parameter info))	749275	an 5f	200	adi
74824460	beryot BCourts at Property (con parameter info)	740270	ala Si	pop	
74924000	beryptice protection of the second se	748270		pop	
7492960-0	beryptiberypteneratesymmetrickey (kio parameter info)	740270	at SD	pop	epx durand star [sha d]
74925100	be up to the product of the former of the fo	746270		100	eck, dword ptr [eop-4]
74025100	beryptiberyptifianizekeyraif (no parameter into)	746270	00 5500	xor.	
74025280	beryptiberyptercrypt ((no parameter into))	748270	02 e8e5a40000	Call	bcrypt:scryptResolveProviders+0x2tic (7483209c)
/4825560	bcryptiBcryptUecrypt (<no 1070)<="" parameter="" td=""><td>74827b</td><td>b7 c9</td><td>Leave</td><td></td></no>	74827b	b7 c9	Leave	
74025050	bcryptibcryptexportkey (kno parameter intos)	748270	08 C21C00	ret	
/4825000	bcrypt:Bcryptimportkey (<no into="" parameter="">)</no>	74827b	bb cc	int	
74825180	bcryptiBcryptimportKeyPair (<no into="" parameter="">)</no>	74827b	bc cc	int	
748264d0	bcryptlBCryptSecretAgreement (<no info="" parameter="">)</no>	748276	bd cc	int	
74826610	bcrypt!BCryptDeriveKey (<no info="" parameter="">)</no>	74827b	be cc		
74826760	bcrypt1BCryptDuplicateKey (<no info="" parameter="">)</no>	74827b	bf cc		
74826920	bcrypt!BCryptDestroyKey (<no info="" parameter="">)</no>	bcrypt	<pre>!BCryptGenRandom:</pre>		
74826a50	bcrypt!BCryptDestroySecret (<no info="" parameter="">)</no>	74827b	c0 8bff	mov	edi,edi
74826b20	bcrypt!BCryptCreateHash (<no info="" parameter="">)</no>	74827b	c2 55	push	ebp
74826fd0	bcrypt!BCryptHashData (<no info="" parameter="">)</no>	74827b	c3 8bec		ebp,esp
74827060	bcrypt!BCryptFinishHash (<no info="" parameter="">)</no>	74827b	c5 83e4f8	and	esp,0FFFFFF8h
74827170	bcrypt!BCryptCreateMultiHash (<no info="" parameter="">)</no>	74827b	c8 51	push	
74827310	bcrypt!BCryptProcessMultiOperations (<no info="" parameter="">)</no>	74827b	c9 56	push	
74827390	bcrypt1BCryptDestroyHash (<no info="" parameter="">)</no>	74827b	ca a100408374		eax,dword ptr [bcrypt!BCryptResolveProviders+0x4e80 (74834000)]
748274a0	bcrypt!BCryptDuplicateHash (<no info="" parameter="">)</no>	74827b	cf 8b7514		esi,dword ptr [ebp+14h]
748276b0	bcrypt!BCryptSignHash (<no info="" parameter="">)</no>	74827b	d2 3d00408374		eax,offset bcrypt!BCryptResolveProviders+0x4e80 (74834000)
74827970	bcrypt!BCryptVerifySignature (<no info="" parameter="">)</no>	74827b	d7 741e		bcrypt!BCryptGenRandom+0x37 (74827bf7)
74827bc0	bcrypt!BCryptGenRandom (<no info="" parameter="">)</no>	74827b	d9 f6401c04		byte ptr [eax+1Ch],4
74827ca0	bcrypt!BCryptDeriveKeyCapi (<no info="" parameter="">)</no>	74827b	dd 7418		bcrypt!BCryptGenRandom+0x37 (74827bf7)
748282f0	bcrypt!BCryptDeriveKeyPBKDF2 (<no info="" parameter="">)</no>	74827b	df 56		
748286b0	bcrypt!BCryptKeyDerivation (<no info="" parameter="">)</no>	74827b	e0 ff7510		dword ptr [ebp+10h]
74828790	<pre>bcrypt!BCryptSetAuditingInterface (<no info="" parameter="">)</no></pre>	74827b	e3 ff750c	push	dword ptr [ebp+0Ch]
748287a0	bcrypt!BCryptHash (<no info="" parameter="">)</no>	74827b	e6 ff7508		dword ptr [ebp+8]
74829510	bcrypt!BCryptGetFipsAlgorithmMode (<no info="" parameter="">)</no>	74827b	e9 ff7014	push	dword ptr [eax+14h]
7482d510	<pre>bcrypt!BCryptRegisterProvider (<no info="" parameter="">)</no></pre>	74827b	ec ff7010	push	dword ptr [eax+10h]
7482d6e0	bcrypt!BCryptUnregisterProvider (<no info="" parameter="">)</no>	74827b	ef 6a28	push	28h
7482d7c0	<pre>bcrypt!BCryptQueryProviderRegistration (<no info="" parameter="">)</no></pre>	74827b	f1 59		
7482d970	bcrypt!BCryptEnumRegisteredProviders (<no info="" parameter="">)</no>	74827b	f2 e8e3b3ffff	call	bcrypt+0x2fda (74822fda)
7482da00	bcryptlBCryptCreateContext (<no info="" parameter="">)</no>	74827b	f7 f7c602000000	test	esi.2
7482db10	bcrypt!BCryptDeleteContext (<no info="" parameter="">)</no>	74827b	fd 742e		bcrvpt!BCrvptGenBandom+0x6d (74827c2d)
7482dc10	bcrypt!BCryptEnumContexts (<no info="" parameter="">)</no>	74827b	ff 837d0800	cmp	dword ptr [ebp+8].0
7482dd00	bcrypt!BCryptConfigureContext (<no info="" parameter="">)</no>	74827c	03 740c		bcrvpt/BCrvptGenBandom+0x51 (74827c11)
7482de00	<pre>bcrypt!BCryptOueryContextConfiguration (<no info="" parameter="">)</no></pre>	748270	05 be0d0000c0	mov	es1.8C000000ph
7482df90	bcryptlBCryptAddContextFunction (<no info="" parameter="">)</no>	74827c	0a 685a190000	nush	195Ab
7482e0d0	bcrypt!BCryptRemoveContextFunction (<no info="" parameter="">)</no>	748270	0f eb6c	imp	bcrvpt!BCrvptGenRandom+0xbd (74827c7d)
7482e200	bcrypt!BCryptEnumContextFunctions (<no info="" parameter="">)</no>	74827c	11 8b5510	mov	edx.dword ntr [eho+10h]
7482e3a0	bcrypt!BCryptConfigureContextFunction (<no info="" parameter="">)</no>	74827c	14 83e6fd	and	esi @FFFFFFDh
7482e4e0	bcrypt!BCryptQueryContextFunctionConfiguration (<no info="" parameter="">)</no>	748270	17 8b4d0c	mov	ecx.dword_ptr [ebp+0Ch]
7482e6b0	bcryptlBCryptAddContextFunctionProvider (<no info="" parameter="">)</no>	74827c	1a 56	nush	esi
7482e810	bcrypt!BCryptRemoveContextFunctionProvider (<no info="" parameter="">)</no>	748276	1b e869370000	call	bcrvpt!BCrvptGetFipsAlgorithmMode+0x1e79 (7482b389)
7482e970	bcrvpt!BCrvptEnumContextFunctionProviders (<no info="" parameter="">)</no>	748270	20 Shf0	mov	esi.eax
7482eb30	bcrypt/BCryptSetContextFunctionProperty (<no info="" parameter="">)</no>	748276	22 85f6	test	esi.esi
7482ecb0	bcrypt!BCryptQueryContextFunctionProperty (<no info="" parameter="">)</no>	748270	24 796a	ins	hcrynt/BCryntGenBandom+BydB (74827c98)
1	competence and a second s	748270	26 6864190000	nush	1964h
		748270	2h eb50	imp	hcrypt!BCryptGenBandom+0xhd (74827c7d)
<000:6		740270	ad shadas	Junt	acy duand at [aba18]



Since this API has been deprecated, I would expect the actors to change this in a future version of the ransomware, perhaps to the newer Cryptography API: Next Generation (CNG), which provides a new API BCryptGenRandom to achieve the same result.

A significant difference between the earlier Ryuk binary and our recent sample is the time it takes to fully encrypt the local disk. The 2018 binary takes close to one hour to encrypt the local disk, while the 2020 version takes less than 10 minutes.

✓ ■ 8d3f68b16f0710f858d8c	10060	11.09	62.23 kB/s	3.9 MB	DESKTOP-FRH\Marco	
🗸 📧 net.exe	10028			1.23 MB	DESKTOP-FRH\Marco	Net Command
🔤 conhost.exe	2620			1.98 MB	DESKTOP-FRH\Marco	Console Window Host
📧 net1.exe	8704			1.22 MB	DESKTOP-FRH\Marco	Net Command
🗙 💶 net.exe	7060			1.2 MB	DESKTOP-FRH\Marco	Net Command
🔤 conhost.exe	10140			1.99 MB	DESKTOP-FRH\Marco	Console Window Host
🗙 💶 net.exe	8944			1.11 MB	DESKTOP-FRH\Marco	Net Command
conhost.exe	9720			2.04 MB	DESKTOP-FRH\Marco	Console Window Host
net1.exe	2864			1.12 MB		
💙 📧 net.exe	6924			1.11 MB	DESKTOP-FRH\Marco	
📧 conhost.exe	5472			2 MB		
net1.exe	7708			948 kB		
💙 📧 net.exe	6440			512 kB	DESKTOP-FRH\Marco	Net Command
🔤 conhost.exe	7692			1.73 MB	DESKTOP-FRH\Marco	Console Window Host
💙 📧 net.exe	4720			512 kB	DESKTOP-FRH\Marco	Net Command
conhost.exe	4932			1.57 MB	DESKTOP-FRH\Marco	Console Window Host
💙 📧 net.exe	9588			516 kB	DESKTOP-FRH\Marco	Net Command
🔤 conhost.exe	7048			1.02 MB	DESKTOP-FRH\Marco	Console Window Host
🕶 📧 net.exe	4944			1.11 MB	DESKTOP-FRH\Marco	Net Command
conhost.exe	9580			1.96		
📧 net1.exe	4852			1.12	E7.1	$2 \cdot 0 0$
💙 📧 net.exe	3604			1.11	5/.1	3.09
🔤 conhost.exe	4204			2		
not1 ovo	2560			540 kP	DISK TOD SPIN MARCO	

Figure 3: 2018 Ryuk 2018 Slower Encryption

The increased encryption speed of the newer Ryuk variant places an extra burden on enterprise security efforts. The reaction time to detect, mitigate, and eradicate Ryuk before major damage is done is significantly limited, and many organizations have been unable to contain the ransomware in time. This occurred in <u>UHS networks</u> and required hospital staffers to shut down computer systems <u>immediately</u> to prevent further machines becoming infected by Ryuk.

Diving Deeper into Ryuk 2020

This particular Ryuk sample

(f8bc1638ec3b04412f708233e8586e1d91f18f6715d68cba1a491d4a7f457da0) has a signed digital certificate which was explicitly revoked by its issuer.

- Serial Number: 0a 1d c9 9e 4d 52 64 c4 5a 50 90 f9 32 42 a3 0a
- Subject: CN = K & D KOMPANI d.o.o

Digital Signature Details			2	x	
General Advanced Digital Signat This digital sign	t ure Information nature is OK.				
Signer information Name: K E-mail: N Signing time: S	& D KOMPANI d.o.o. ot available Gaturday, September 26	, 2020 5:30:5 View Certific	9 PM cate		Figure
Countersignatures Name of signer: DigiCert Timestamp	Responder	E-mail ad Not availa Details	Time: Satur		
			OK		

4: 2020 Ryuk Revoked Certificate

When Ryuk begins executing, it duplicates itself and dumps this copy into the same directory with a randomly generated 8 character name. However, the file name always ends with "... lan.exe". These duplicate files help to start multiple threads. Ryuk utilizes a list of hardcoded strings to search for and stop specific running processes (Figure 1). It then tries to inject itself into additional processes.

i f8bc1638ec3b04412f708233e8586e1d91f18f	10/5/2020 6:33 PM	Application	377 KB
GUSicTuVqlan.exe	10/5/2020 6:33 PM	Application	377 KB
TDdkcGWMzlan.exe	10/5/2020 6:33 PM	Application	377 KB
WKgOBEziwlan.exe	10/5/2020 6:33 PM	Application	377 KB
L"C:\\Users\\NTK\\Desktop\\New folder\\GUS	icTuVqlan.exe"		

&L"\"C:\\Users\\NTK\\Desktop\\New folder\\GUSicTuVqlan.exe\" 8 LAN"

&L"C:\\Users\\NTK\\Desktop\\New folder\\GUSicTuVglan.exe"

&L"\"C:\\Users\\NTK\\Desktop\\New folder\\GUSicTuVqlan.exe\" 8 LAN"

Figure 5: Droppers

Ryuk next begins executing certain command line tools to achieve some of its devastating effects; in particular, it tries to prevent user recovery by attempting to delete the Volume Shadow copies by leveraging cmd.exe /c 'WMIC.exe shadowcopy delete' . This is followed with a cmd.exe /c 'vssadmin.exe Shadows /all /quiet" and cmd.exe /c 'bcdedit /set {default} recoveryenabled No & bcdedit /set {default}'.



Figure 6:

Services that are not stopped

An **icacls.exe** is created in the Windows WoW directory, which gives the group **Everyone** full permissions to the drives on the system so that Ryuk has everything it needs to encrypt all drives.



Figure 7b: icacls permissions granted

Extracting the Executable from Memory

To avoid detection, the malware uses various evasion techniques like self injection. Ryuk uses this technique by allocating memory in which it writes a PE file. After this, it calls **VirtualProtect** to change the execution permissions on the section.

A fast way to extract the executable from memory is to run the binary in a debugger and set a breakpoint at the location of the allocated memory. For this, I use **x32dbg**, and set a breakpoint on **VirtualAlloc**. A thing to note is that when setting a breakpoint for VirtualAlloc you should follow the **jmp** routine into **Kernelbase** to get the base address of the newly allocated region and set the breakpoint on the return. Once the debugger has run, the breakpoint will hit. Follow the **EAX** register to the memory dump section to view if the **MZ** magic is present.



When the process is run, it will hit the breakpoint **VirtualAlloc** and in the **EAX** is the newly allocated virtual memory section to begin loading a copy of itself into this section. Following the **EAX** to the memory dump shows that the memory has been allocated for loading. When continuing the process, the dump window begins to populate with data as it hits the breakpoint that is set several times. Once it is confirmed that the binary has been fully loaded into this section, the binary data can be dumped for inspection.

00 00 05 F F 6 6 7 6 7 7 7 7 7 7 7 7 7 7 7
00 05 5 6 7
72 65 73 74 72 65 74 74 75 65 74 74 75 65 74 74 75 65 74 74 75 65 74 74 75 65 74 74 74 75 74 75 74 74 74 75 74 74 74 75 74 74 74 75 74 74 75 74 74 74 75 74 <td< td=""></td<>
69 67 6E 65 70 72 69 63 74 28 00 igned.restrict(65 00 00 00 00 02 64 65 65 74 new
$ \begin{array}{c} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
00 00 3D 3D 00 00 21 3D 00 00 25 85 0 00 06 F 70 ,=, i=., i=., i=., i=., i=., i=., i=
b) $72 \ 61 \ 74 \ 67 \ 72 \ 00 \ 00 \ 00 \ 00 \ 00 \ 00 \ 0$
00 00 <td< td=""></td<>
00 00 3C 00 00 00 28 29 00 00 3E 00 00 3E 30
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$
$\begin{array}{cccccccccccccccccccccccccccccccccccc$
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$
00 00 60 76 62 74 61 62 74 61 62 70 00 66 77 00 00 call'. 'typeof'. 63 61 66 67 00 00 66 66 74 72 69 62 ucta''static 67 75 61 72 64 27 00 00 00 62 61 73 64 67 00 00 60 65 63 74 72 66 67 00 60 66 64 65 66 67 00 60 66 66 66 67 00 <t< td=""></t<>
63 61 6C 62 67 70 65 67 66 27 00 <td< td=""></td<>
67 75 61 72 64 27 00 00 00 60 73 74 72 69 Eguard' stri 67 75 61 72 64 27 00 00 60 76 62 64 65 73 g' vbase de 74 72 75 63 74 67 22 04 65 63 74 72 75 63 74 67 72 00 00 00 60 64 65 63 74 67 20 64 65 63 74 67 20 64 65 63 74 67 20 64 65 73 74 72 75 63 74 67 20 69 74 72 ros 11 constructor 10
67 27 00 <td< td=""></td<>
74 72 20 66 67 67 76 65 65 74 67 72 r <
75 63 74 6F 72 27 70 00 00 60 64 65 66 61 75 uctor''defa 72 20 64 65 62 72 70 00 00 60 73 63 74 67 72 20 63 61 65 61 15 constructor 1 72 20 63 66 67 74 75 63 74 67 72 20 63 74 67 72 00 00 00 00 60 76 65 63 74 67 r r dettor''vett 72 03 64 67 74 67 74 67 74 65 74 65 74 67 74 67 74 67 74 65 74 67 74 67 74 67 74 65 63 74 67 74 65 74 67 72 77 63 74 6
6C 74 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 63 1t constructor 72 20 64 65 6C 65 74 69 6E 67 20 64 65 73 74 72 r deleting dest 75 63 74 6F 72 27 00 00 00 60 73 63 61 6C 61 losure''scal 72 20 64 65 73 74 72 75 63 74 6F 72 20 69 74 r constructor i 65 72 61 74 6F 72 27 00 00 00 60 76 65 63 74 6F erator''vect 72 20 64 65 73 74 72 75 63 74 6F 72 20 69 74 r constructor i 72 61 74 6F 72 27 00 00 00 60 76 65 63 74 6F rator''vect 72 20 76 62 61 73 65 20 63 6F 6E 73 74 72 75 63 r vbase constru 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 60 76 tor iterator''vect 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 60 76 tor iterator''vect 75 63 74 6F 72 20 69 74 65 72 61 73 65 20 63 6F 6E 73 74 72 75 63 r vbase constru 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 60 76 tor iterator''vect 74 6F 72 20 69 74 65 72 61 74 6F 72 70 00 00 60 65 68 20 76 65 ent map''eh v 63 74 6F 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 ctor constructo 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 00 00 60 65 iterator'' 768 20 76 65 63 74 6F 72 20 64 65 73 74 72 75 63 h vector destru 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 65 72 61 onstructor iter 74 6F 72 27 00 00 60 63 6F 70 79 20 63 6F 6E 73 tor''copy con 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''copy con 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''copy con 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''copy con 73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''copy con 73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''copy con 73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''copy con 73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tor''dt returnin 74 66 72 27 00 00 60 70 6C 61 63 65 60 56 E 74 20 72 65 27 delete closure 75 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 delete closure 76 63 61 6C 20 76 66 74 61 62 6C 65 20 63 61 6C 6C 74 '. new[] dele 65 74 65 75 85 D0 00 00 00 00 00 60 66 66 66 73 75 72 65 27 delete closure 76 63 74 6F 72 20 69 74 65 72 72 70 00 00 60 60 65 68 74 6F 72 20 69 74 r constructor i 65 72 61 74
The second seco
75 63 74 6F 72 27 00 00 00 60 75 65 63 74 6F uctor''vect 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 6F r constructor i 72 61 74 6F 72 27 00 00 60 76 65 63 74 6F r ator' vect 72 20 64 65 73 74 72 75 63 74 6F 72 20 69 74 65 r destructor it 72 61 74 6F 72 20 69 74 65 72 61 73 65 20 63 6F 6E 73 74 72 75 63 r vbase constru 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 00 60 76 65 63 74 6F rator''vect 75 63 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 60 76 65 61 63 65 60 irtual displace 65 6E 74 20 60 61 70 27 00 00 60 66 68 20 76 65 ent map'. eh v 63 74 6F 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 constructo 74 6F 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 constructo 74 6F 72 20 63 74 6F 72 20 64 65 73 74 72 75 63 h vector constructo 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 00 00 00 60 65 irtual displace 68 20 76 65 63 74 6F 72 20 64 65 73 74 72 75 63 h vector vbase 67 6E 73 74 72 75 63 74 6F 72 20 69 74 65 72 61 onstructor iter 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 65 72 61 onstructor iter 74 6F 72 70 00 06 06 3 6F 70 79 20 63 6F 6E 73 tor''copy con 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tructor closure 'udt returnin' 67 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C ocal vftable'.' 67 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C ocal vftable'.' 67 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C ocal vftable'.' 67 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C ocal vftable'.' 67 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C ocal vftable'.' 67 63 61 6C 20 76 66 74 61 62 6C 65 73 75 72 65 cr '. new[] dele 65 58 5D 00 00 00 60 6F 60 6E 67 3 75 72 65 74 '. new[] dele 65 74 65 72 50 00 00 60 67 66 67 3 75 72 65 67 70 00 00 60 70 6C 61 63 65 6D 65 6E 74 '. new[] dele 65 74 65 78 50 20 63 6C 6F 73 75 72 65 27 00 00 coal vftable'.' 66 74 65 72 70 00 00 60 70 6C 61 63 65 6D 65 6E 74 20 64 65 6C 65 74 '. new[] dele 65 74 65 72 20 69 74 65 72 20 69 74 65 72 20 69 74 y constructor i 65 74 65 72 20 69 74 65 72 61 74 6F 72 20 69 74 y constructor i 65 74 6F 72 20 69 74 65 72 61
$\begin{array}{cccccccccccccccccccccccccccccccccccc$
72 20 64 65 74 65 63 74 65 74 67 67 74 67 74 67 74 67 74 67 74 67 74 67 67 67 65 63 74 67 77 72 75 63 74 67 77 77 72 76 62 67 74 67 77 77 72 76 63 67 67 77 77 72 76 63 67 67 77 77 72 76 63 66 67 74 77 75 63 74 67 72 70 00 00 00 00 00 66 66 66 67 74 72 76 63 74 67 72 70 00 00 00 00 66 66 66 67 74 72 75 63 74 67 72 70 00 60 65 67 65 67 <td< td=""></td<>
72 61 74 6F 72 20 00 <td< td=""></td<>
74 6F 72 20 69 74 66 72 27 00 60 76 tor iterator'. 69 72 74 75 61 6C 20 64 69 73 70 6C 61 63 65 6D irtual displace 65 6E 74 20 63 6F 72 74 72 75 63 74 67 72 20 63 66 67 74 72 75 63 74 67 72 74 72 75 63 74 67 72 70 00 00 00 00 00 65 65 63 74 67 72 70 00 00 00 00 00 65 65 74 67 74 72 75 63 74 67 72 70 00 60 65 77 70 60 66 73 75 72 61 71 72 75 63 74 67
69 72 74 75 61 6C 20 64 69 73 70 6C 61 63 65 6D intual displace 65 6E 74 20 60 61 70 00 00 60 68 20 76 65 ent map''eh v 20 69 74 65 72 61 74 67 72 74 72 75 63 74 67 20 ctor constructor 20 69 74 65 72 61 74 67 72 70 00 00 00 60 65 tor iterator'' 68 20 76 65 63 74 67 72 20 66 67 72 63 64 67 20 63 67 67 72 64 67 72 62 67 tor''copy con onstructor closure 'udt returnin 'cotal 'udt returnin'.' 65 63 61 62 62 63 </td
63 74 67 20 63 66 67 20 63 66 67 20 63 66 67 63 74 65 63 74 66 72 61 74 67 72 63 74 67 72 61 74 67 72 63 74 67 72 63 74 67 72 63 74 67 72 63 74 67 72 63 74 67 72 63 74 67 72 63 74 67 72 63 64 67 72 70 60 65 63 74 67 72 70 60 65 63 74 67 72 63 64 67 72 70 60 65 77 63 74 67 72 63 64 67 72 63 66 67 77 72 65 77 tructor 70 70 60 64 64 62 66 65
20 69 74 65 72 61 74 6F 72 27 00 00 00 00 00 60 65 iterator'' 68 20 76 65 63 74 6F 72 20 64 65 73 74 72 75 63 h vector destru 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 60 65 68 20 76 65 63 74 6F 72 20 76 62 61 73 65 20 63 h vector vbase 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 65 72 61 onstructor iter 74 6F 72 27 00 00 60 63 6F 70 79 20 63 6F 6E 73 tor''copy con 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tructor closure 00 00 60 75 64 74 20 72 65 74 75 72 6E 69 6E 67 'udt returnin 27 00 60 45 48 00 60 52 54 54 49 00 00 00 60 6C '.'EH.'RTTI' 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C 73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C 73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 5ructor closure 27 00 20 6E 65 77 58 5D 00 00 20 64 65 6C 65 74 'ew[]dele 65 58 5D 00 00 00 60 67 6C 61 63 65 6D 65 6E 74 20 63 61 6C 6C 73 74 65 78 5D 00 00 60 67 6C 61 63 65 6D 65 6E 74 20 64 65 6C 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 delete closure 00 00 60 70 6C 61 63 65 6D 65 67 42 06 64 65 6C 'placement de 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 delete closure 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 77 58 57 00 00 60 60 61 61 62 61 67 65 64 20 76 65 63 74 65 76 50 20 63 6F 6E 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 68 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 68 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 72 61 74 6F 72 27 00 00 00 60 6D 61 61 6E 61 67 65 63 74 6F 72 20 69 74 6F 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 6F 73 74 72 75 63 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 00 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 74 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69
$\begin{array}{cccccccccccccccccccccccccccccccccccc$
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$
6F 6E 73 74 72 75 63 74 6F 72 20 69 74 65 72 61 onstructor iter 74 6F 72 70 00 60 66 70 79 20 63 6F 62 73 tor''copy con 74 72 75 63 74 67 72 65 77 tor''copy con 74 72 75 63 74 67 72 65 77 tructor closure 00 00 60 75 64 74 20 25 54 49 00 00 66 66 74 61 62 65 67 00 66 66 74 61 62 66 65 66 66 67 75 75 63 67 67 75 75 65 57 75 75 63 67 67 70 60 66 67 73 75 72 65 57 <
74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 27 tructor closure 00 00 60 75 64 74 20 72 65 74 75 72 6E 69 6E 67udt returnin 27 00 60 45 48 00 60 52 54 54 49 00 00 00 60 6C 'EH.RTTI' 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 65 74 '. new[] dele 65 58 5D 00 00 00 60 6F 6D 6E 69 20 63 61 6C 6C 74 sig' placemen 20 64 65 6C 65 74 65 20 63 6C 6F 73 75 72 65 27 delete closure 20 00 60 70 6C 61 63 65 6D 65 6E 74 20 64 65 6Cplacemen de 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 72 61 74 6F 72 27 00 60 00 60 60 61 61 6E 61 67 65 63 74 6F 72 20 69 74 7 constructor i 65 72 61 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 00 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 7 constructor i 65 72 61 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 68 20 76 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 68 20 76 65 68 20 76 65 63 74 6F 72 20 69 74 9 constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65
00 00 60 75 64 74 20 72 65 74 75 72 6E 69 6E 67udt returnin 27 00 60 45 48 00 60 52 54 54 49 00 00 00 60 6C '.`EH.`RTTI` 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 6F 63 61 6C 20 76 66 74 61 62 6C 65 27 00 60 6C coal vftable'. 77 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 structor closur 70 02 06 65 77 58 5D 00 00 20 64 65 6C 65 74 '. new[] dele 65 58 5D 00 00 00 60 6F 6D 6E 69 20 63 61 6C 6C e[] ommi cal 73 69 67 27 00 00 60 70 6C 61 63 65 6D 65 6E 74 sig' placemen 20 64 65 6C 65 74 65 20 63 6C 6F 73 75 72 65 27 delete closure 00 00 60 70 6C 61 63 65 6D 65 6E 74 20 64 65 6C placement de 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 ete[] closure'. 00 00 60 6D 61 6E 61 67 65 64 20 76 65 63 74 6F managed vect 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 r constructor i 65 72 61 74 6F 72 20 09 74 65 72 61 74 6F 72 27 00 00 ctor iterator' mana 65 64 20 76 65 63 74 6F 72 20 64 65 73 74 72 75 63 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 20 69 74 6F 72 20 63 6F 70 eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 67 74 67 eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 erator' eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 v constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 erator' eh vector 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 erator' eh vector 65 72 61 74 6F 72 27 00 76 65 63 74 6F
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$
73 74 72 75 63 74 6F 72 20 63 6C 6F 73 75 72 65 Structor Closur 73 70 70 6E 65 77 58 5D 00 00 20 64 65 6C 65 74 '. new[] dele 65 58 5D 00 00 00 60 6F 6D 6E 69 20 63 61 6C 6C e[] omni cal 73 69 67 27 00 00 60 70 6C 61 63 65 6D 65 6E 74 sig' placemen 20 64 65 6C 65 74 65 20 63 6C 6F 73 75 72 65 27 delete closure 00 00 60 70 6C 61 63 65 6D 65 6E 74 20 64 65 6C placement de 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 ete[] closure'. 00 00 60 6D 61 6E 61 67 65 64 20 76 65 63 74 6F managed vect 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 r constructor i 65 72 61 74 $\frac{6F}{72}$ 27 00 00 00 60 6D 61 6E 61 67 erator' managed vect 63 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 v constructor i 63 74 6F 72 20 69 74 65 72 61 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 72 27 00 00 00 60 65 68 20 76 65 ca 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 72 27 00 00 00 60 65 68 20 76 65 ca 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 72 27 00 00 00 60 65 68 20 76 65 ca 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 72 27 00 00 00 60 65 68 20 76 65 ca 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{6E}$ 72 27 00 00 00 60 65 68 20 76 65 ca 63 64 65 70 70 70 70 70 00 ctor y baca conv
65 5B 5D 00 00 60 6F 6D 6E 69 20 63 61 6C 6C e[]*omni cal 73 69 67 27 00 00 60 67 6C 61 63 65 6C 67 73 75 72 65 74 sig'placement 20 64 65 6C 65 74 65 60 65 67 75 72 65 27 delete closure 00 00 60 70 6C 61 63 66 65 67 20 64 65 62 placement delete closure 00 00 60 61 6E 61 67 65 63 74 65 63 74 65 63 74 67 65 63 74 67 65 63 74 67 65 63 74 67 72 75 63 74 67 72 65 63 74
73 69 67 27 00 00 60 70 6C 61 63 65 6D 65 6E 74 sig'. placement 20 64 65 6C 65 74 65 20 63 6C 6F 73 75 72 65 27 delete closure 00 00 60 70 6C 61 63 65 6D 65 6E 74 20 64 65 6Cplacement de 65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 ete[] closure'. 00 00 60 6D 61 6E 61 67 65 64 20 76 65 63 74 6Fmanaged vect 72 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 r constructor i 65 72 61 74 $\frac{6F}{22}$ 27 00 00 00 60 6D 61 6E 61 67 erator'mana $\frac{65}{64}$ 20 76 65 63 74 6F 72 20 64 65 73 74 72 75 ed vector destr 63 74 6F 72 20 69 74 65 72 61 74 6F 72 27 00 00 ctor iterator'. 00 00 60 65 68 20 76 65 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{22}$ 72 70 00 00 00 60 65 68 20 76 65 erator''eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{22}$ 72 70 00 00 00 60 65 68 20 76 65 erator''eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 $\frac{6F}{22}$ 27 00 00 00 60 65 68 20 76 65 erator''eh vector co
00 00 60 67 68 68 67 20 64 65 65 67 74 75 63 74 66 66 67 74 72 75 63 74 67 72 20 63 67 67 74 77 63 74 67 72 20 69 74 77 74 77 74 77 74 77 75 63 74 67 72 20 64 65 74 77 75 63 74 67 72 75 64 67 <td< td=""></td<>
65 74 65 58 5D 20 63 6C 6F 73 75 72 65 27 00 00 etc.] closure'. 00 00 60 61 6E 61 67 65 64 20 76 65 63 74 6F `managed vect'. 72 20 63 6F 6E 73 74 6F 72 20 69 74 r constructor i 65 72 61 74 6F 72 20 60 60 61 6E 61 67 r constructor i 65 72 61 74 6F 72 20 64 65 73 74 72 75 63 74 67 72 70 00 00 60 61 6E 61 67 entor'`managed vector 63 74 67 72 20 64 67 72 75 63 74 67 72 75 61 <td< td=""></td<>
72 20 63 66 67 74 72 75 63 74 66 73 74 77 75 63 74 66 73 74 77 75 63 74 66 73 74 72 75 63 74 67 72 20 69 74 r constructor i 65 72 61 74 65 63 74 67 72 20 64 65 73 74 72 75 ed vector destr 63 74 67 72 20 64 65 73 74 72 75 ed vector destr 63 74 67 72 20 64 65 73 74 72 75 ed vector destr eh vector destr eh vector co 79 20 63 65 63 74 67 72 20 63 67 0 eh vector co eh v
65 72 61 74 <u>6F</u> 72 27 00 00 60 61 6E 61 67 erator'`mana <u>65</u> 64 20 76 65 63 74 F 72 20 64 65 73 74 75 ed vector destr 63 74 6F 72 20 64 65 72 70 00 00 ctor iterator'`mana 63 74 6F 72 20 64 65 73 74 75 ed vector destr 00 00 60 65 68 20 76 65 74 6F 72 20 63 67 0``eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 6F 72 20 69 74 y constructor i 63 72 61 <
65 64 20 76 65 74 66 72 70 64 73 74 75 64 Vector destribution 63 74 66 72 61 74 67 72 70 00 00 ctor iterator'. 00 00 60 65 68 20 76 65 74 67 72 20 63 67 0. . `eh vector co 79 20 63 67 65 63 74 67 72 20 63 67 0. . `eh vector co 79 20 63 67 65 73 74 72 75 63 74 67 72 20 69 74 y constructor i 65 72 61 74 <u>67 72 70 00 66 68 20 76 65</u>
00 00 60 65 68 20 76 65 63 74 6F 72 20 63 6F 70 eh vector co 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 y constructor i 65 72 61 74 6F 72 27 00 00 00 60 65 68 20 76 65 erator' eh v 63 74 6F 72 20 76 62 61 73 65 20 76 65 erator' eh v
79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 y constructor 1 65 72 61 74 6F 72 27 00 00 00 65 68 20 76 65 erator' eh v 63 74 6F 72 00 76 62 61 73 65 20 63 65 erator' eh v
63 74 6E 72 20 76 62 61 73 65 20 63 6E 70 79 30 ctar vhase carv
05 / 07 / 20 / 0 02 02 / 5 05 20 05 0F / 0 / 5 20 CLUI VDASE COPY
63 6F 6E 73 74 72 75 63 74 6F 72 20 69 74 65 72 constructor ite
6E 69 74 69 61 6C 69 7A 65 72 20 66 6F 72 20 27 nitializer for
00 00 60 64 79 6E 61 6D 69 63 20 61 74 65 78 69 dynamic atex
20 27 00 00 00 00 60 76 65 63 74 6F 72 20 66 6F 72 t destructor to
70 79 20 63 6F 6E 73 74 72 75 63 74 6F 72 20 69 py constructor
74 65 72 61 74 6F 72 27 00 00 60 76 65 63 74 6F terator' vect
73 74 72 75 63 74 6F 72 20 69 74 65 72 61 74 6F structor iterat
72 27 00 00 00 00 60 60 61 6E 61 67 65 64 20 76 r' managed
72 75 63 74 6F 72 20 69 74 65 72 61 74 6F 72 27 ructor iterator
00 00 60 6C 6F 63 61 6C 20 73 74 61 74 69 63 20 local static
74 68 72 65 61 64 20 67 75 61 72 64 27 00 6F 70 thread guard'.0 65 72 61 74 6F 72 20 22 22 20 00 00 00 00 20 54 erator ""
79 70 65 20 44 65 73 63 72 69 70 74 6F 72 27 00 ype Descriptor'
00 00 20 42 61 73 65 20 43 6C 61 73 73 20 44 65 Base Class D
61 73 65 20 43 6C 61 73 73 20 41 72 72 61 79 27 ase Class Array
00 00 20 43 6C 61 73 73 20 48 69 65 72 61 72 63 Class Hierar
00 00 20 43 6F 6D 70 6C 65 74 65 20 4F 62 6A 65 Complete Obj

Figure 9: Dump Memory To File

22	Follow in Disassembler		Г
	Follow in Dump		00
P	Dump Memory to File		P ¹
	Comment	;	
Þ	Find Pattern	Ctrl+B	
Ę.	Switch View		
M	Find references to region		
	Allocate memory		00
	Free memory		00
60	Add virtual module		sk
6	Go to	•	ac
	Set Page Memory Rights		
•	Memory Breakpoint	+	ac
A	Сору	•	po

The next step is to right-click on the memory dump and follow the dump in the memory map. This brings you to where the dump has been allocated in memory, and from here you can dump the memory to file. However, as shown in Figure 9, notice that the dumped memory does not have a valid PE header; we have to modify the header so the PE (Figure 10) can work in the tool of your choice.

📓 File Edit S	Search	Vie	w	Anal	sis	Тоо	ls V	/indo	ow ł	Help													F	s) F	File	Edit	Se	arch	n Vi	iew	An	alysi	s To	ools	Win	dow	He	lр											
🗋 🚵 🕶 🐻			1	f	• 1	6	~	W	lindo	ws (ANSI)	\sim	h	ex	1	\sim								2	- ,				a)	•	+ +	16		~	Wine	lows	(AN	SI)		\sim	hex		~					
pe_f8bc1638	8ec3b0	4412	f708	233e	8586	ie1d	91f18	f671	5d68	cba1	a491	d4a71	457d	a0_0	01D00	000.1	.bin	1					FE	à r	be_f	8bc1	638e	c3b(0441	12f70	823	3e85	86e1	d91f	18f6	715d	i8cba	a1a49	1d4a	7f45	7da0	_001	D000	0.bi	n				
Offset(h)	00 (01 ()2	03	04	05	06	07	08	09	OA (ово	oc o	DO	E 01	F	De	200	de	d	text			of:	fse	et (h) (00	01	02	03	04	1 05	5 06	5 07	08	09	0A	0B	0C	0D	0E	OF	D	eco	ded	te	xt	
000000000	85 1	02 1	E :	20	8D	4F	24	03	C8 (83	79 (04 (0 8	вo	1 75	5	Ò	ð~	.0	\$.1	Èfy.	.<.u		00	000	0000		4D	5A	90	00	03	8 00	00	00	04	00	00	00	FF	FF	00	00	М	z			ÿ	ÿ
000000D0	05 (17 1	17	38 3	EB	03	03	41	04 1	3B	C6 (DF 4	17 F	4 3	3 CI	1		. G8		.A	.;Ε. Ρυό/	Gðfi		00	000	0010	Ì	B8	00	00	00	00	0 00	00	00	40	00	00	00	00	00	00	00		• • •	•••	.0.	• • •	
000000F0	5F !	50 8	D	50	FF	8D	48	FF	F7 1	D2	48 (03 0	E O	3 0	3 23	3	P	P.P	ÿ.	Hÿ-	÷ÒH.	1.Ã#		00	000	0020		00	00	00	00	00	000	00	000	00	00	00	00	C8	00	00	00					È	
00000100	CA 2	23 0	2	3B	C1	75	99	6A	04	68	00	30 0	0 0	0 5	3 61	A	Ê#	Â;i	Áu	™j	.h.0			00	000	0040) (0E	lF	BA	0E	: 00	B4	09	CI	21	B8	01	4C	CD	21	54	68		·°.	· ^ •	Í!,	.LÍ	!Th
00000110	5C 2	24 1	18	33	D2	89	8B 44	24	24	8B	CB 1	85 E	D 7	4 3	3 81	B	•⊻ ∖\$.30	ò:	D\$	< сэ\ \$<Ё	3A01 .it3<		00	000	0050		69 74	73	20 62	70 65	20	2 61	67	72 6F	61 20	6D	20 6E	63 20	61 44	6E 4F	6E	6F 20	1	s p	rog ru	ram ni	n D	nno 005
00000130	5C 2	24 (5C :	23	D8	4D	85	DB	74	19	8B (27 2	B 4	4 2	4 50	2	\\$	\$1#0	ØM		t.«Ç	+D\$\		00	000	0070		6D	6F	64	65	2E	I OI	0	07	24	00	00	00	00	00	00	00	n	ode		.ş.		
00000140	3B 1 00 1	DO 1 EB (13	DF	83 06	FA 88	3C 01	72 41	05 8 46	83 42	FA : 85 1	BE 1 ED 1	6 0	5 C 7 8	6 01 B 50		;Đ)s.: 	fú ^	<r< td=""><td>.fú> FB_i</td><td>v.E.</td><td></td><td>00</td><td>000</td><td>0080</td><td></td><td>E5</td><td>4D</td><td>88 2D</td><td>C2</td><td>Al</td><td>20</td><td>E</td><td>91</td><td>A1</td><td>20</td><td>E6</td><td>91</td><td>Al B4</td><td>2C</td><td>E6</td><td>91</td><td>å</td><td>M^Ä ó_∖</td><td>i,æ</td><td>11</td><td>æ`i</td><td>, æ `</td></r<>	.fú> FB_i	v.E.		00	000	0080		E5	4D	88 2D	C2	Al	20	E	91	A1	20	E6	91	Al B4	2C	E6	91	å	M^Ä ó_∖	i,æ	11	æ`i	, æ `
00000160	24	18 (F	B7	47	06	OF	B7	4F (14	85 (20 1	4 3	6 8	3 C1	7	\$.				0À	t6f9		00	000	DOAO		62	23	BB	91	A2	2 20	E	91	14	B2	06	91	AO	2C	E6	91	b	#»`	¢,æ		.,	,æ`
00000170	2C (03 0	F	BB	7C	24	5C	8B	51 1	F8	48 1	BB 3	0	3 I	3 81	B		.Ï< ; _:	\$ ~D		QøH<	1.00		00	000	DOBO		14	B2	38	91	AO	20	E	91	52	69	63	68	Al	2C	E6	91	•	*8 `	, æ	'Ri	chi	,æ'
00000190	42 4	16 4	D	75	F7	8B	44	24	5C	83	C1 :	28 8	5 C	0 7	5 D	7	BF	Mu	÷<	D\$	\fÁ(Àu		00	000			5D	87	6E	5F	00		00		00	00	00	00	EO	00	02	21	i	+n		. PE	à	
000001A0	8B	7C 2	4	10	8B	B7	80	00	00	00	03 1	F3 8	9 7	4 2	4 14	4	<	\$.	< 1	€.	ó	tt\$.		00	000	DOEC) (0B	01	0C	00	00	17	00	00	00	F6	01	00	00	00	00	00				ö		
00000180	10 8	36 (38 I	18	BB	06	03	EB	03	C3 :	89	ff :	24 5	6C 8	37	D 00		< E . <	 ø<	At). ë.	APYI ĉD\$	\$ <i \f}.</i 		00	000	00F0		B0 00	27	00	00	00	0 10			000	30	00	00	00	00	00	10				0		
000001D0	00 -	74 4	F	BB	74	24	20	8B	08 (85	C9 .	74]	E 7	9 1	C 81	в	.t				Ét			00	000	0110	0 (06	00	00	00	00	0 00	00	00	00	40	02	00	00	04	00	00				@		
000001E0	47 :	BC (F I	87 88	C9 RB	8B	44	38	78 2	2B	4C :	38 J	.0 8	B 4	4 38	B	G<	<1 	Ē<	D8:	x+L8 ra /	. < D8		00	000	0120	0 (00	00	00	00	02	2 00	40	0 01	00	00	10	00	00	10	00	00		• • •	@	• • •	• • •	
00000200	02 (03 0	3	50	57	FF	D6	89	45	00	83 (C5 (14 8	B 4	4 24	4		ÄP	Wÿ	Ö	E.fÅ	.< D\$																											
00000210	5C 8	33 0	0	04	89	44	24	5C	83	7D	00	00 1	5 B	98	B 74	4	\ <i>f</i> .	FÀ.	*D	\$\:	f}	u ¹ < t																											
00000220	24 . 8B '	14 8 7C 2	3B -	46 . 10	20 8B	83 EB	2B	14 6F	34	83	24 . BF 1	14 8 A4 (00 0	0 7	0 00	2	ş. <1	. < E S	∵t. ∢ë	# +o	‱tŞ. 4f/я	Au‡																											
00000240	OF 8	34 7	A	00	00	00	8B	97	AO (00	00 (00 0	3 D	38	9 54	4	• "					.Ó‰1																											
00000250	24 !	5C 8	D ·	4A RB	04 32	8B 8D	01	89	4C 2	24	14 (F8 (B5 0	0 0	FS	4 81	D	ş١			. 5.	L\$ fàa	A.,																											
00000270	44 2	24 2	20	74	60	6A	02	8B	D8	5A	OF I	B7 ()F 4	Be	6 81	в	DŞ	τ	ij		ØZ.	.Kf<																											
00000280	Cl (66 0	1 1	E8	0C	66	83	F8	OA '	74	06	66 8	3 F	8 0	3 75	5	Áf	EÁè			.t.f	fø.u																											
00000290 000002A0	24 °	75 1	1	B1 1	El	FF	OF	00	00	8B	C5 (27 0 Cl H	18 1	0 6	6 01	1	 \$u	.ay 1	áÿ		,⊥e .∢ÅÁ	i;D4 è.f.																											
000002B0	04 3	31 H	В	DF	66	3B	C2	75	A 0	81 3	E1 1	FF (F 0	0 0	0 66	6	.1			Âu	áÿ	1																											
00000200	01 2 5C 2	C 3	81 I	03	FA	85	DB	75	B1 8	8B	5C :	24] 8D 4	88	B 5	4 24	4	1	1.1	ú)	.Uu:	±<\\$ ⊤ŝ\	. < T \$																											
000002E0	89 4	1C 2	4	14	85	CO	OF	85	77 1	FF	FF I	FF 8	B 7	C 2	4 10	D	%L	L\$	À		wÿÿÿ	< \$.																											
000002F0	OF I	87 4	17	06	OF	B7	4F	14	85 0	CO	OF I	84 E	32 0	0 0	0 00	0	• •	G.			À.,,	f																											
00000310	0D 24 :	5C (F	B6	8F	10	00	00	8B	45	00 :	33 0	:9 8	B 5	5 00	2	с с \$\				enj) (E.3	É <u.< td=""><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></u.<>																											
00000320	41 (C1 E	8	lD	Cl	EA	lE	23	C1 :	23	Dl (BB 4	D 0	0 0	1 E9	9	АÁ	iè.i	Áê	• #J	Á#Ñ<	M.Áé																											
00000330	1F 8 6A (35 ()4 5	E I	75 6A	18	85	D2 C9	75 58	07 (0F	6A 44	08 : F0 1	SE 6 EB 2	C 8	1 E 5 D	B 05	5	• 1.	.Au .^i		.Ou ÉX	∙j•^ .Dāë	j.ë. . ò:																											
00000350	17 8	85 0	9	75	04	6A	10	EB	15 (85	D2 '	75 0	B 8	5 C	9 74	4		Éu			Òu	Ét																											
00000360	18 H	BE 8	0		00	00	EB	11	85 0	C9	75 0	05 6	A 2	0 5	E EF	B	.*4	€.	 ÷v	ë	…Éu. s∴⊽	j^ė																											
00000380	74 (06 8	81 (CE	00	02	00	00	8D	44	24 :	28 5	50 8	B 4	5 E8	B	۰J t.	.î			.D\$(P∢Eè																											
00000390	56 1	FF 1	15 1	EC	03	C3	50	FF	54 :	24	3C 1	85 0	0 0	F 8	4 FI	D	Vÿ	/uì	.Ã	Pÿ.	Ţ\$<	À. "S																											
000003A0 000003B0	FC 1 FF 1	FF I FF 8	B	8B 77	44 28	24 6A	5C 00	83 6A	00	28 6A	85 (FF (CO 0 D3 H)F 8 73 F	5 5 F 5	7 FI	4	üÿ VV	7ÿ<] /< W	DŞ (1)	\f1 .i	Α(Α .iÿ.	óÿTS																											
000003C0	3C :	33 0	:0	40	50	50	53	FF	D6 (83	7C :	24 6	50 0	0 7	4 70	2	<3	BÀ@	PP	Sÿ	Öfļ\$	•.t																											
000003D0 000003E0	83 74	7F 1	1C	79	14	76	8B	4F 64	78 (8B	03 69	CB 1	BB 4	9 2	88	5 C(D	f. ti	. ifv		× O: t.d.	x.Ek	A																											
000003F0	83	54 2	4	5C	00	03	FB	85	C0	74	51	BB	5 0	0 0	3 F3	3	fd	1\$\		û	ÀtQ<	uć																											
00000400	33 1	02 0	F	BE	06	C1	CA	OD	03 1	DO	46	80 1	EF	FO	0 75	5	3Ò).%	i. Á	Ê.	.ĐF€	~ÿ.u																											
00000410	C7 (2 8	9	44	24	5C	3B	9B 41	18	72	DO I	EB J	.5 0	FB	7 11	7	пэ Ç.	515 %D		;A	.rĐë	A.01																											
00000430	83 1	FA I	F	74	17	8B	41	1C	FF	74	24	68 H	F 7	4 2	4 68	в	fú	iÿt		Α.	ÿt\$h	ÿt\$1																											
00000440	8D (5D)	5B 9	33	8B 24	04 48	18 C3	03	C3 EC	FF 1	D0 64	59 : Al	59 8 30 0	BC	3 5 0 0	F 5H	3	··· 1 I	∢ IfÄl	нÃ	.A	yÐYY .d:0	< A																											
00000460	55 3	56 8	BB	40	OC	57	89	4C	24	18	8B	70 0	C E	9 8	A 00	D	UV	7< @	. W	ħL.	\$. <p< td=""><td>.éŠ.</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></p<>	.éŠ.																											
00000470	00	12 0	BB	46	30	33	C9	8B	SE 2	2C	8B :	36 8	9 4	4 2	4 14	4		< F		É<	^,<6	aD\$																											
00000480	EB :	10 3	33	FF	85	DB	74	1F	8B	6C	24	14 8	A O	4 2	F CI	1	¢В ё.	.3ÿ.	Û	t.	<1\$.	š./ <u>1</u>																											
000004A0	C9 (DD 3	C	61	OF	BE	CO	7C	03 1	83	C1 1	E0 0	3 C	8 4	7 31	в	É.		.*	ÀI	. fÁà	ÈG;																											
000004B0 000004C0	FB '	72 H	22	8B 89	6C 7C	24	10	8B	94 2 FF	2A 74	20 : 31 :	33 I 88 3	B 8	B 7 3 F	C 21	8 3	ür	re∢: Â≞	1\$ 18	• <	D* 3 Vtl∢	iUk * (3⊽																											
000004D0	EA 8	33 (:0	04	89	44	24	10	OF I	BE	45	00 00	:1 C	FO	D 03	3	êf.	FÀ.	\$D		. %E .	ÁÏ.																											
000004E0	F8 4	45 8	80	7D	FF	00	75 2P	FO	8D 0	04	OF :	3B 4	4 2	4 1	8 74	4	øE	E€}		uð	;	D\$.t																											
00000500	D2 (OF 8	15	6B	FF	FF	FF	33	CO	5F	5E !	5D 5	B 8	3 0	4 10	D	ò.	k	ÿŸ	ÿ3	À ^]	[fÄ.																											
00000510	C3 8	BB	14	24	10	8B	44	16	24	8D	04 !	58 0	FB	7 0	C 10	D	Ã<			D.	\$x																												
00000520	8B 4	14] 00 (03	1C	8D 00	04	88 04	5B 00	00	10	03 (FF 1	SZ E	BD	0 F	D 51	A. D	< D			×	A	eUM2																											
00000540	00 0	00 0	00	00	00	00	40	00	00	00	00	00 0	0 0	0 0	0 00	D				0.																													

Figure 10: Fix the Headers

This particular binary is straightforward to modify. Open up your favorite Hex editor, load the file, highlight everything before the MZ and delete it. Sometimes, just deleting extra bytes will not work if a blob of memory has corrupted magic bytes. In that scenario, you can copy a known good header and add it to the corrupted PE header to make a valid PE.



Figure 11: Three Memory Dump Files

If you follow the process from the beginning, the breakpoint will hit **VirtualAlloc** additional times. I've dumped the memory with the techniques shown above to show why Ryuk's encryption on a system is so fast:

					[Yes	s No	
0)9	:1	4:	:3′	1	?		
Sat	/e	R	eset	St	an			Figure
Round: 4					00:26:88	1		
Round: 3					00:16:07	1	DESKTOP-ERH.	м
Round: 2					00:34:48	r	DESKTOP-FRH DESKTOP-FRH	
Round: 1					01:40:21		DESKTOP-FRH.	. <u>Л</u> .Д
					arco		RH	М
180c1638ec3b04412f708 sTtzMglySlan.exe	10216	64.72 1.97	111.79 M	99.07 MB 1.73 MB	DESKTOP- Marco	01	:40:21	
iii) xjZphbeUClan.exe	3240	2.00		1.74 MB	DESKTOF Marco	00:	:34:48	
wTSSoGDIQIan.exe	5028	2.28		1.75 MB	DESKTOP Marco	00:	:16:07	
	50.40	2 44		1.65 MR	DESKTOP			

Conclusion

The FBI has stated that Ryuk Ransomware actors have been paid over <u>61 million dollars</u>. With Ryuk attacks crippling organizations, this number will soon surpass the 100 million mark if it hasn't done so already.

However, guarding against the ransomware menace in general and Ryuk in particular is not complicated with the <u>proper protection</u> in place: The techniques used by these cybercriminals are well-understood and relatively simple. The weaknesses they exploit are organizations' inability to detect and remediate at <u>speed</u>, but this is a problem that can be and has been <u>solved</u>.

Meanwhile, as analysts, it's important that we keep up with the latest developments and techniques deployed by adversaries. At SentinelOne, we track the ever-changing variants of Ryuk to understand the latest capabilities added to this ransomware family. In this post, we have detailed how Ryuk has evolved to increase its speed of encryption and the methods it uses for evasion. In a future post, we will cover Ryuk's network layer and the many artifacts collected during our analysis process.

Samples

SHA256: f8bc1638ec3b04412f708233e8586e1d91f18f6715d68cba1a491d4a7f457da0 SHA1: c3fa91438850c88c81c0712204a273e382d8fa7b

SHA256: 7e28426e89e79e20a6d9b1913ca323f112868e597fcaf6b9e073102e73407b47 SHA1: 5767653494d05b3f3f38f1662a63335d09ae6489

MITRE ATT&CK

Command and Scripting Interpreter T1059 Native API T1106 Application Shimming T1546.011 Process Injection T1055 Masquerading T1036 Virtualization/Sandbox Evasion T1497.001 Deobfuscate/ Decode Files T1140 Obfuscated Files or Information T1027 System Time Discovery T1124 Security Software Discovery T1518.001 Process Discovery T1057 File and Directory Discovery T1083 System Information Discovery T1082 Archive Collected Data T1560 Encrypted Channel T1573