

Descriptografando Strings em Malwares

Ronaldo Pinheiro de Lima

Ministério Público Federal

GTS-17 - São Paulo - 14/05/2011

www.crimesciberneticos.com

Apresentação

- ❑ Servidor Público do MPF
- ❑ Procuradoria da República de Bauru / SP
- ❑ Grupo de Combate aos Crimes Cibernéticos PR-SP
- ❑ Investigações de Phishings/Malwares
- ❑ Pesquisador independente de segurança
- ❑ Blog: www.crimesciberneticos.com

Agenda

Introdução

Criptografia de Strings em Malwares

Descriptografia através de:

- Dump de memória
- Debugger e Breakpoint
- Engenharia Reversa
- Script Python + Immunity Debugger

Introdução

Criptografar: modificar um texto para impedir sua compreensão pelos que não conhecem seus caracteres ou convenções. (Houaiss)

Sinônimos: ofuscar*, encriptar

Reverso: descriptografar, desencriptar

Inglês: decrypt, de-obfuscation, decoding

Criptografia de Strings em Malwares

- ❑ **Objetivos:** esconder no executável informações relevantes , dificultar o rastreamento e a ER
- ❑ Strings são visíveis no *disassembly*
- ❑ Ex.: URLs, e-mails, arquivos, banco de dados, etc
- ❑ Função de descriptografia está no próprio EXE
- ❑ Descriptografia "*on the fly*"

Criptografia de Strings em Malwares

```
decrypt(string encrypted){  
    ...  
    return decrypted;  
}  
envia_email(){  
    de = "vitima@mail.com";  
    para = "s7mZqeRlp8n01ZxdTqaGb3zLqo4RsBzQ1";  
    assunto = "Dados capturados Santander";  
    texto = "Agência 999 CC 999999 Senha 123456";  
    mail(de, decrypt(para), assunto, texto);  
}  
main(){  
    if(dadoscapturados==true) envia_email();  
}
```

Pseudocódigo de envio de e-mail com string criptografada.

Criptografia de Strings em Malwares

Text string	
UNICODE	"nvbd5afdn+bu56q6n6KdJafEn6A6AaMbn+Pd0afL5+H656cD5+fd5+q0n65dnaKu5+v6AH"
UNICODE	"nvbd5afdn+bu56q6n6KdJafEn6A6Aaq0n+5dnaf6n65d5aPEn+bdn6qQnvPd5aK95+funDqL
UNICODE	"nPqd56M2nvAd5+M2na5dnaMDn+Pd56q0nvqG56M2nvPunDq0"
UNICODE	"n6KdnDM2naAdJafEnvqLnDqLn+A"
UNICODE	"n+KunDMknPqd56M2nvPunDqQnvPunaq0n6K"
UNICODE	"nPqd56M2nvAd5+M2na5dnaMDn+Pd56q0nvqG56M2nvPunDq0"
UNICODE	"n65dA6fGnD5d0+qGn+vdn6M2nDKd5aMkn+5d0H"
UNICODE	"n6AuJDFGnDbdn6qLnvPd56qQn+bGnDqLn65undqQnvPd0H"
UNICODE	"n6AuJDFGnD5d0+Mkn6Ad5aAGn+Pu56fGn+vd0aM2"
UNICODE	"n6AuJDFGnDAun+A6nvqd0afGn+bGnDqLn65undqQnvPd0H"
UNICODE	"n65dA6fGnaAdJafEnvqG56M2nvPunDq0nDAdnaf6n6AdJaMknvq"
UNICODE	"n65dA6fGnaAdJafEnvqL5DqLn65u56M2n+b"
UNICODE	"n6AuJDFGnbPd0+Mbn+PGnDqLn65undqQnvPd5afGn+bu5+qQnvq"
UNICODE	"n6AuJDFGnD5u5DqdnD5d0afEnvbGnDqLn65undqQnvPd0H"
UNICODE	"n6AuJDFGnaqd5aMDnvqu5+02n6wdnafhn+bd56q0nvq"
UNICODE	"n65dA6fGnD5d0+qGn+vdn6M2nDqdJaMkn+bdA6qQn+Ad5aqGn+P"
UNICODE	"n6AuJDFGnDAdnaf6n+5u5+qQn+5d5aM2nbqu5DqLn6Kd5aq6n+bd0H"
UNICODE	"n+KunDMknPqd56M2nvPdn+qQn6Kd0Dq0"
UNICODE	"n+bu5DMdnPqd56M2n6Ku5DM2"
UNICODE	"n+bu56f6n+vd0aq0n6Aunafhn+bh5DqLnv5dnafGn6c2nDMkn+vd56q05+wu5Dq0n6Kd5aKE
UNICODE	"n+5d5aMbn6wd0+P6n+Pd0aq9n+b"
UNICODE	"n+KunDMknPqd56M2nvPdn+qQn6Kd0Dq0n6K"
UNICODE	"n+bu5DMdnPqd56M2n6Ku5DM2"
UNICODE	"Cu0"

Criptografia de Strings em Malwares

```
LEA ECX,DWORD PTR SS:[EBP-18]  
MOV EDX,NetEmpre.004D3DC0  
MOV EAX,ESI  
CALL NetEmpre.004D32A0  
MOV EAX,DWORD PTR SS:[EBP-18]  
CALL NetEmpre.00409768  
TEST AL,AL  
JNZ NetEmpre.004D39B2  
LEA ECX,DWORD PTR SS:[EBP-1C]  
MOV EDX,NetEmpre.004D3C58  
MOV EAX,ESI  
CALL NetEmpre.004D32A0  
MOV EAX,DWORD PTR SS:[EBP-1C]  
CALL NetEmpre.00409768  
TEST AL,AL  
JNZ NetEmpre.004D39B2  
LEA ECX,DWORD PTR SS:[EBP-20]  
MOV EDX,NetEmpre.004D3E04  
MOV EAX,ESI  
CALL NetEmpre.004D32A0  
MOV EAX,DWORD PTR SS:[EBP-20]  
CALL NetEmpre.00409768  
TEST AL,AL  
JNZ NetEmpre.004D39B2  
LEA ECX,DWORD PTR SS:[EBP-24]  
MOV EDX,NetEmpre.004D3E50  
MOV EAX,ESI  
CALL NetEmpre.004D32A0  
MOV EAX,DWORD PTR SS:[EBP-24]  
CALL NetEmpre.00409768  
TEST AL,AL  
JNZ NetEmpre.004D39B2
```

ASCII "GpfSGN9nTMbsRtCWP6KWS79lPt9X

ASCII "GpfSLqbEH4zNKrn4RtTkR6zXP6La

ASCII "GpfSLqbEH4zNKrn4RtTkR6zXP6La

ASCII "GpfSLqbEH4zNKrn4RtTkR6zXP6La

Descriptografia de Strings através de **Dump de Memória**

- Cópia da memória utilizada pelo malware em um determinado momento
- Permite visualizar as strings descriptografadas que estão na memória
- **Ferramentas:**
 - Standalone: Userdump, LordPE
 - Forense de memória: Volatility
 - Plug-in para debugger: OllyDump

Descriptografia de Strings através de Dump de Memória

system.exe (Trojan.Win32.VB.ajfm)

Disassembly	Text string
MOV EDX,system.00404708	UNICODE "012258464142447919694746476D5255445C5C555C1F5
MOV EDX,system.0040485C	UNICODE "008260716A0568737B17646309080301006A770701037
MOV EDX,system.00404910	UNICODE "012658464142447919694746476D5255445C5C555C1F5
MOV EDX,system.00404A50	UNICODE "009858464142447919694746476D5255445C5C555C1F5
MOV EDX,system.00404C34	UNICODE "0152584641420D6C193147461E205444555555D52504
MOV EDX,system.00404D74	UNICODE "005258464142447919694746476D5255445C5C555C1F5
MOV EDX,system.00404B24	UNICODE "0052584641420D6C193147461E214351545C43535E1F5
MOV EDX,system.00404B9C	UNICODE "0044584641420D6C19244250542642535F17535F5C1F5
MOV EDX,system.00404E14	UNICODE "0106584641424479196947464730421E524B515454425
MOV EDX,system.00404EF8	UNICODE "0120584641424479196947464730421E524B515454425
MOV EDX,system.00405008	UNICODE "0090584641420D6C193147461E30595F405F5153585D1
MOV EDX,system.004050CC	UNICODE "0146584641424479196947464730421E524B515454425
MOV EDX,system.00405200	UNICODE "0262584641420D6C19244250542642535F505E4454435
MOV EDX,system.004050CC	UNICODE "0146584641424479196947464730421E524B515454425
MOV EDX,system.0040541C	UNICODE "0128584641424479196947464730421E524B515454425
MOV EDX,system.004055F8	UNICODE "0130584641424479196947464730421E524B515454425
MOV EDX,system.0040570C	UNICODE "0062584641420D6C193147461E214351545C43535E414
MOV EDX,system.0040552C	UNICODE "0054584641420D6C19244250542642535F4942595C541
MOV EDX,system.004057D8	UNICODE "0262584641420D6C19244250542642535F505E4454435
MOV EDX,system.004050CC	UNICODE "0146584641424479196947464730421E524B515454425
MOV EDX,system.00403C00	UNICODE "024860405A445E2753340D62610F7E7C757D721E000A6
MOV EDX,system.004059F4	UNICODE "004063575957543716125F411072111A105F425F5C11"
MOV EDX,system.00403B94	UNICODE "00226470796D792C402972555E"
MOV EDX,system.00405A54	UNICODE "004810657D776506166E105D5F20505C594351545E431

Descriptografia de Strings através de Dump de Memória



```
C:\>system.exe

C:\>userdump -k system.exe system.exe_dump
User Mode Process Dumper (Version 8.1.2929.5)
Copyright (c) Microsoft Corp. All rights reserved.
Dumping process 1408 (system.exe) to
C:\system.exe_dump...
The process was dumped successfully.

C:\>
C:\>strings -q -o -n 5 system.exe_dump > strings_system.txt

C:\>
C:\>_
```

Utilização do userdump para copiar a memória utilizada pelo malware system.exe (Trojan.Win32.VB.ajfm)

Descriptografia de Strings através de **Dump de Memória**

□ Strings capturadas:

SOFTWARE\Microsoft\MSSQLServer\Client\ConnectTo

Provider=SQLOLEDB.1;Password=**h8090100**;

User ID=**salaodefes13**;Initial Catalog=salaodefes13;

Data Source=**dbsq0012.whservidor.com**;

BradaFisicoTabela

117B01787A1F7808717E037D08047A781A79740E1E60016E7C0

00226470796D792C402972555E

Descriptografia de Strings através de Dump de Memória

DB5Q0012.sal...o.TBL_NovoBdn Summary					
	Registro	localizador	agencia	conta	senhanet
▶	1	234201010816	2342	0101081-6	
	2	013300974331	0133	0097433-1	
	3	00361080601	0036	108060-1	
	4	□□ 522422	□□	52242-2	
	5	407522422	407	52242-2	
	6	1286338753	1286	33875-3	
	7	319554348	3195	5434-8	
	8	154500992437	1545	0099243-7	
	9	14755217857	1475	521785-7	
	10	2863120987	2863	12098-7	
*	NULL	NULL	NULL	NULL	NULL

Banco de dados utilizado pelo malware system.exe (Trojan.Win32.VB.ajfm)

Descriptografia de Strings através de **Debugger e Breakpoint**

- Debuggers permitem acompanhar a execução do malware linha a linha
- **Assembler-Level Debuggers:** OllyDbg, Immunity Debugger, IDA Pro, WinDbg
- Localizar a função de descriptografia, colocar um breakpoint no retorno e verificar o resultado
- Como???

Descriptografia de Strings através de Debugger e Breakpoint

❑ 1- Localizar a função de descriptografia

LEA ECX,DWORD PTR SS:[EBP-18]	ASCII "GpfSGN9nTMbsRtCWP6KWS79lPt9X
MOV EDX,NetEmpre.004D3DC0	
MOV EAX,ESI	
CALL NetEmpre.004D32A0	→ 004D32A0
MOV EAX,DWORD PTR SS:[EBP-18]	
CALL NetEmpre.00409768	
TEST AL,AL	
JNZ NetEmpre.004D39B2	
LEA ECX,DWORD PTR SS:[EBP-1C]	ASCII "GpfSLqbEH4zNKrn4RtTkR6zXP6La
MOV EDX,NetEmpre.004D3C58	
MOV EAX,ESI	
CALL NetEmpre.004D32A0	→ 004D32A0
MOV EAX,DWORD PTR SS:[EBP-1C]	
CALL NetEmpre.00409768	
TEST AL,AL	
JNZ NetEmpre.004D39B2	
LEA ECX,DWORD PTR SS:[EBP-20]	ASCII "GpfSLqbEH4zNKrn4RtTkR6zXP6La
MOV EDX,NetEmpre.004D3E04	
MOV EAX,ESI	
CALL NetEmpre.004D32A0	→ 004D32A0
MOV EAX,DWORD PTR SS:[EBP-20]	
CALL NetEmpre.00409768	
TEST AL,AL	
JNZ NetEmpre.004D39B2	
LEA ECX,DWORD PTR SS:[EBP-24]	ASCII "GpfSLqbEH4zNKrn4RtTkR6zXP6La
MOV EDX,NetEmpre.004D3E50	
MOV EAX,ESI	
CALL NetEmpre.004D32A0	→ 004D32A0

OllyDbg - NetEmpresa.exe (Trojan-Banker.Win32.Banbra.vhu)

Descriptografia de Strings através de **Debugger e Breakpoint**

- 2- Breakpoint na função para localizar o retorno

004D329B	00	DB 00
004D329C	00	DB 00
004D329D	8D40 00	LEA EAX,DWORD PTR DS:[EAX]
004D32A0	55	PUSH EBP
004D32A1	8BEC	MOV EBP,ESP
004D32A3	83C4 E8	ADD ESP,-18
004D32A6	53	PUSH EBX
004D32A7	56	PUSH ESI
004D32A8	57	PUSH EDI
004D32A9	33DB	XOR EBX,EBX
004D32AB	895D E8	MOV DWORD PTR SS:[EBP-18],EBX
004D32AF	895D 5C	MOV DWORD PTR SS:[EBP-14],EBX

EBP=0012FDF4
Local calls from 004D3839, 004D3858, 004D3877, 004D3896, 004D38B5, 004D38D4

0012FD0C	004D6D92	RETURN to NetEmpre.004D6D92 from NetEmpre.004D32A0
0012FD10	0012FE04	Pointer to next SEH record
0012FD14	004D7221	SE handler
0012FD18	0012FDF4	
0012FD1C	004352A0	NetEmpre.004352A0
0012FD20	00E21958	
0012FD24	00000000	
0012FD28	00000000	
0012FD2C	00000000	

RETURN to NetEmpre.004D6D92

Breakpoint at NetEmpre.004D32A0

Descriptografia de Strings através de **Debugger e Breakpoint**

- 3- Breakpoint no retorno para verificar o resultado

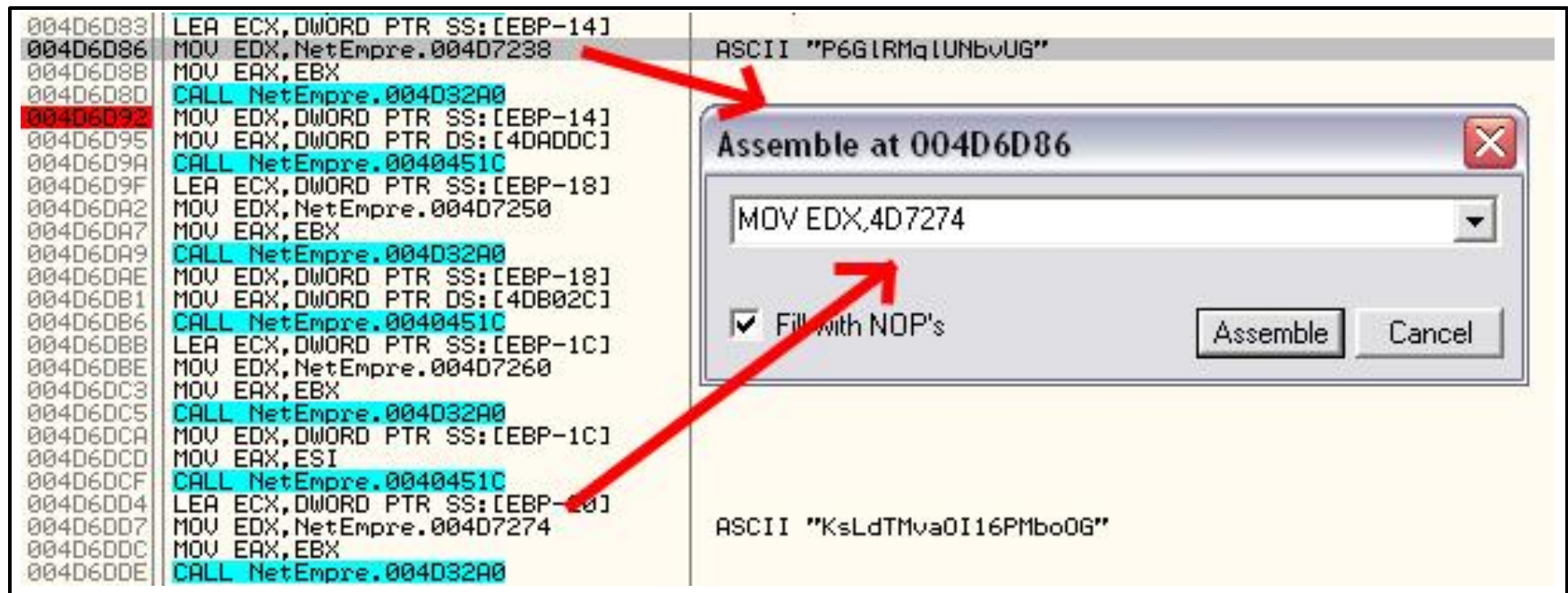
004D6D83	. 8D4D EC	LEA ECX,DWORD PTR SS:[EBP-14]	
004D6D86	. BA 38724D00	MOV EDX,NetEmpre.004D7238	ASCII "P6GIRMqLUNbvUG"
004D6D8B	. 8BC3	MOV EAX,EBX	
004D6D8D	. E8 0EC5FFFF	CALL NetEmpre.004D32A0	
004D6D92	. 8B55 EC	MOV EDX,DWORD PTR SS:[EBP-14]	
004D6D95	. A1 DCAD4D00	MOV EAX,DWORD PTR DS:[4DADDC]	
004D6D9A	. E8 7DD7F2FF	CALL NetEmpre.00404510	
004D6D9F	. 8D4D EC	LEA ECX,DWORD PTR SS:[EBP-14]	

Stack SS:[0012FDE0]=00E12CC0, (ASCII "dd/mm/yyyy")

- Para descriptografar as outras strings manualmente é só alterar o Offset do parâmetro da função (endereço da string criptografada)
- No OllyDbg clique na linha desejada e <Space>

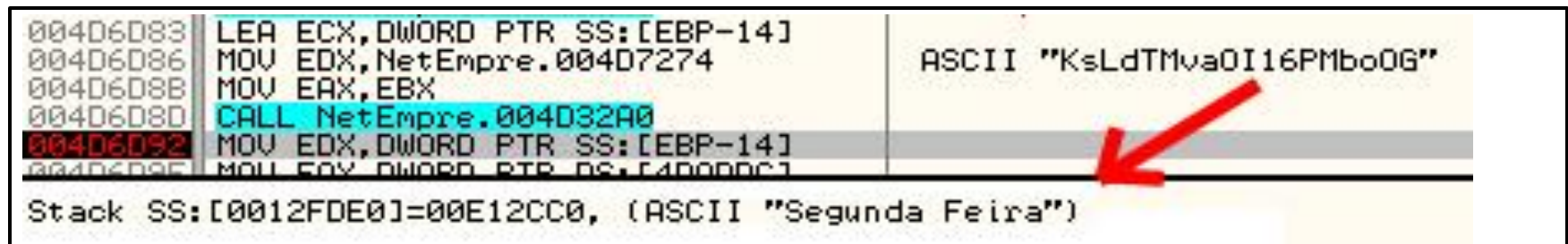
Descriptografia de Strings através de Debugger e Breakpoint

- ❑ Alterar a instrução para descobrir outros resultados



The screenshot shows a debugger window with assembly code on the left and a memory dump on the right. A red arrow points from the instruction `MOV EDX, NetEmpre.004D7238` to a dialog box titled "Assemble at 004D6D86". The dialog box contains the instruction `MOV EDX, 4D7274` and a checkbox labeled "Fill with NOP's" which is checked. The "Assemble" button is highlighted. The memory dump on the right shows the ASCII string `"KsLdTMvaOI16PMboOG"`.

Address	Instruction	Comment
004D6D83	LEA ECX, DWORD PTR SS:[EBP-14]	
004D6D86	MOV EDX, NetEmpre.004D7238	ASCII "P6GIRMq1UNbvUG"
004D6D8B	MOV EAX, EBX	
004D6D8D	CALL NetEmpre.004D32A0	
004D6D92	MOV EDX, DWORD PTR SS:[EBP-14]	
004D6D95	MOV EAX, DWORD PTR DS:[4DADDC]	
004D6D9A	CALL NetEmpre.004D451C	
004D6D9F	LEA ECX, DWORD PTR SS:[EBP-18]	
004D6DA2	MOV EDX, NetEmpre.004D7250	
004D6DA7	MOV EAX, EBX	
004D6DA9	CALL NetEmpre.004D32A0	
004D6DAE	MOV EDX, DWORD PTR SS:[EBP-18]	
004D6DB1	MOV EAX, DWORD PTR DS:[4DB02C]	
004D6DB6	CALL NetEmpre.004D451C	
004D6DBB	LEA ECX, DWORD PTR SS:[EBP-1C]	
004D6DBE	MOV EDX, NetEmpre.004D7260	
004D6DC3	MOV EAX, EBX	
004D6DC5	CALL NetEmpre.004D32A0	
004D6DCA	MOV EDX, DWORD PTR SS:[EBP-1C]	
004D6DCD	MOV EAX, ESI	
004D6DCF	CALL NetEmpre.004D451C	
004D6DD4	LEA ECX, DWORD PTR SS:[EBP-20]	
004D6DD7	MOV EDX, NetEmpre.004D7274	ASCII "KsLdTMvaOI16PMboOG"
004D6DDC	MOV EAX, EBX	
004D6DE0	CALL NetEmpre.004D32A0	



The screenshot shows a debugger window with assembly code on the left and a memory dump on the right. A red arrow points from the instruction `MOV EDX, NetEmpre.004D7274` to the memory dump. The memory dump shows the ASCII string `"KsLdTMvaOI16PMboOG"`.

Address	Instruction	Comment
004D6D83	LEA ECX, DWORD PTR SS:[EBP-14]	
004D6D86	MOV EDX, NetEmpre.004D7274	ASCII "KsLdTMvaOI16PMboOG"
004D6D8B	MOV EAX, EBX	
004D6D8D	CALL NetEmpre.004D32A0	
004D6D92	MOV EDX, DWORD PTR SS:[EBP-14]	
004D6D95	MOV EAX, DWORD PTR DS:[4DADDC]	

Stack SS:[0012FDE0]=00E12CC0, (ASCII "Segunda Feira")

Descriptografia de Strings através de

Engenharia Reversa

- ❑ Engenharia Reversa para entender a função de descriptografia
- ❑ Após entendê-la, reescrevê-la na linguagem preferida
- ❑ Debugger facilita o trabalho
- ❑ Não se perder nas instruções *Assembly*
- ❑ Focar em blocos de códigos, o que representam em linguagem de Alto-Nível
- ❑ Processo demorado, com a prática se torna mais rápido

Descriptografia de Strings através de Engenharia Reversa

pernet.exe (Trojan.Win32.VB.aoyw)

2.213 chamadas para a função de descriptografia

nDfL0+AGnDKL0+PLna5LnaA0naKGJa02nPqG5aPdnDvL5602




G_DB_USUARIO_AVISO

□ Utilizava chave criptográfica

Cript: nDfL0+AGnDKL0+PLna5LnaA0naKGJa02nPqG5aPd...

KEY: w/sDbk2VKcUy5nJTA0paP8xXqMIifS1BH3Z+vjYNRr...

Descriptografia de Strings através de Engenharia Reversa

- Posição do caractere na KEY é utilizada para realizar vários cálculos com constantes
- **+400** linhas Assembly  **45** linhas Python

Algumas strings encontradas:

```
https://www2.bancobrasil.com.br/aapf/  
C:\avenger.txt  
\GbPlugin\bb.gpc  
http://vivaxmotos.com/data/c_c_s.gif  
https://internetbanking.caixa.gov.br/  
senhaConta  
Cadastro_Computador Travou Browser
```


Descriptografia de Strings através de **Script Python + Immunity Debugger**

- ❑ Immunity Debugger: OllyDbg + Python API
- ❑ Executa PyCommands e Scripts Python
- ❑ Automatizar a descriptografia sem precisar entender toda a função
- ❑ O malware não precisa ser executado
- ❑ **Essencial:** Localizar a função de descriptografia, entender quais parâmetros são passados e onde é salvo o retorno (string em texto simples)

Descriptografia de Strings através de Script Python + Immunity Debugger

004D6D83	. 8D4D EC	LEA ECX,DWORD PTR SS:[EBP-14]	
004D6D86	. BA 38724D00	MOV EDX,NetEmpre.004D7238	ASCII "P6GIRMq1UNbvUG"
004D6D8B	. 8BC3	MOV EAX,EBX	
004D6D8D	. E8 0EC5FFFF	CALL NetEmpre.004D32A0	
004D6D92	. 8B55 EC	MOV EDX,DWORD PTR SS:[EBP-14]	
004D6D95	. A1 DCAD4D00	MOV EAX,DWORD PTR DS:[4DADDC]	
004D6D9A	. E8 7DD7F2FF	CALL NetEmpre.0040451C	
004D6D9F	. 8D4D EC	LEA ECX,DWORD PTR SS:[EBP-14]	

Stack SS:[0012FDE0]=00E12CC0, (ASCII "dd/mm/yyyy")



LEA ECX, [EBP-14] ; endereço da pilha é atribuído ao ECX
MOV EDX, NetEmpre.004D7238 ; string criptografada p/ EDX
CALL NetEmpre.004D32A0 ; chamada da função
MOV EDX, [EBP-14] ; retorno da função vem na pilha

- ❑ Localizar todas as chamadas da função (cross references), encontrar os endereços das strings criptografadas e forçar a descriptografia pelo malware

```
import immlib

def main(args):
    j=0
    ret = 0x12FDE0
    imm = immlib.Debugger()

    table = imm.createTable('Strings Descriptografadas
                             NetEmpresa', ['Num', 'Endereco', 'Criptografada',
                             'Descriptograda'])

    refs = imm.getXrefFrom(0x4D32A0)
    for ref in refs:
        addr = 0

        for i in range (1,5):
            op = imm.disasmBackward(ref[0], i)
            instr = op.getDisasm()
            if instr.startswith('MOV EDX,'):
                addr = op.getImmConst()
                break
```

Continua...


```
if addr != 0:
    imm.setReg('ECX',ret)
    imm.setReg('EDX',addr)
    imm.setReg('EIP',ref[0])
    imm.stepOver()

    enc = imm.readString(addr)
    dec = imm.readString(imm.readLong(ret))

    j = j+1
    table.add(' ', ['%d' % j, '0x%X' % addr, '%s' % enc,
                    '%s' % dec])

return "OK"
```

Salvar em: C:\...\Immunity Debugger\PyCommands\decstring.py

Executar: !decstring



Perguntas?

Obrigado!

Ronaldo Pinheiro de Lima

email: ronaldoplima@yahoo.com.br
blog: www.crimesciberneticos.com
twitter: [crimescibernet](https://twitter.com/crimescibernet)

