



Catching the silent whisper: Understanding the Derusbi family tree

Micky Pun, Eric Leung, Neo Tan

Virus Bulletin 2015

Agenda

- What is Derusbi
- Background
- Variants of Derusbi
- Technical Analysis

What is Derusbi





- DLL
- Remote Access Trojan
- Relies on other malware to load or plant on a system
- Resides on a system by imitating legitimate software DLLs (OfficeUt32.dll, Office32.dll, Update.dll...etc) during static file header scanning
- Limited amount of samples (The number of samples since 2008 till today are still in the hundreds)

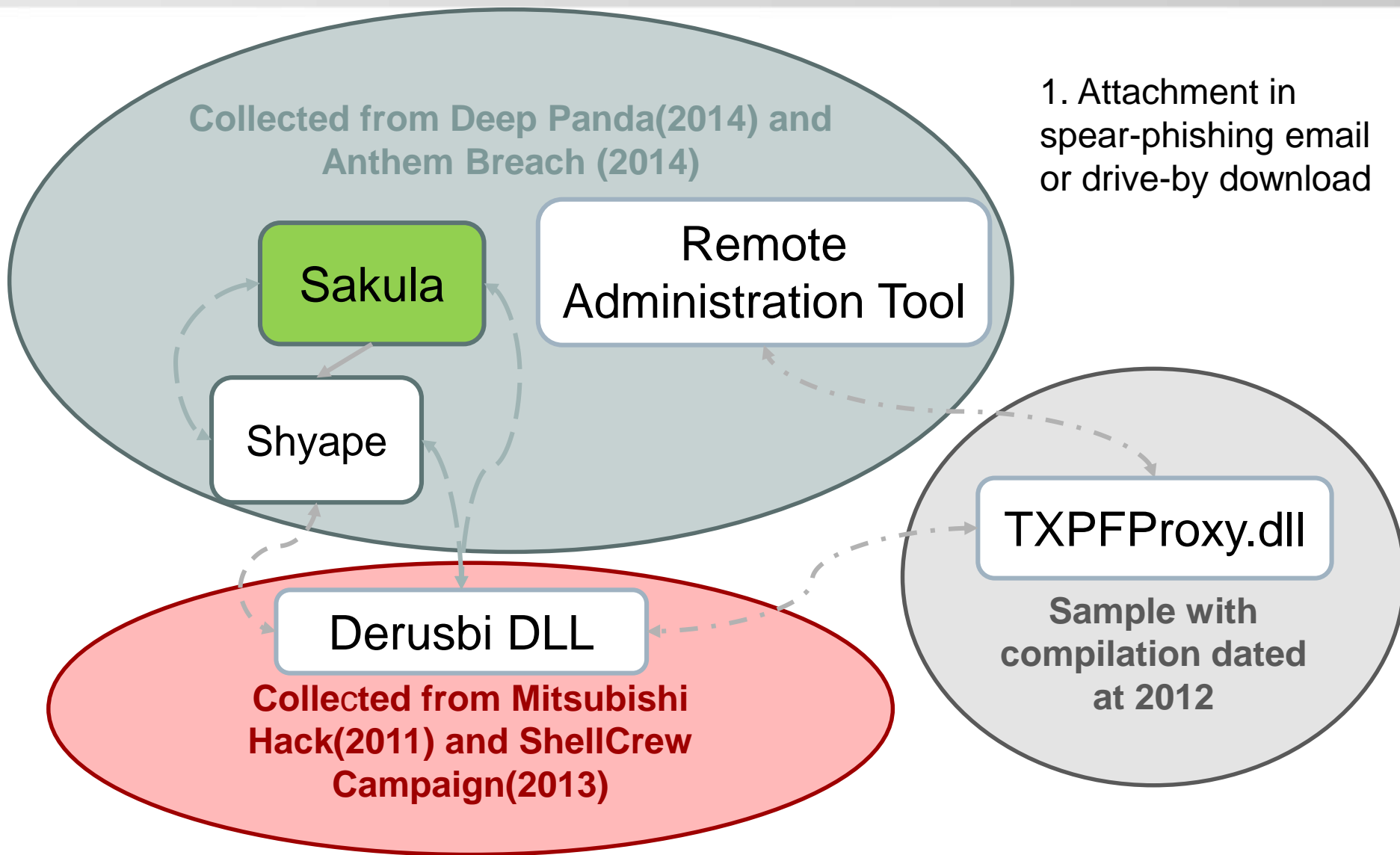
Background



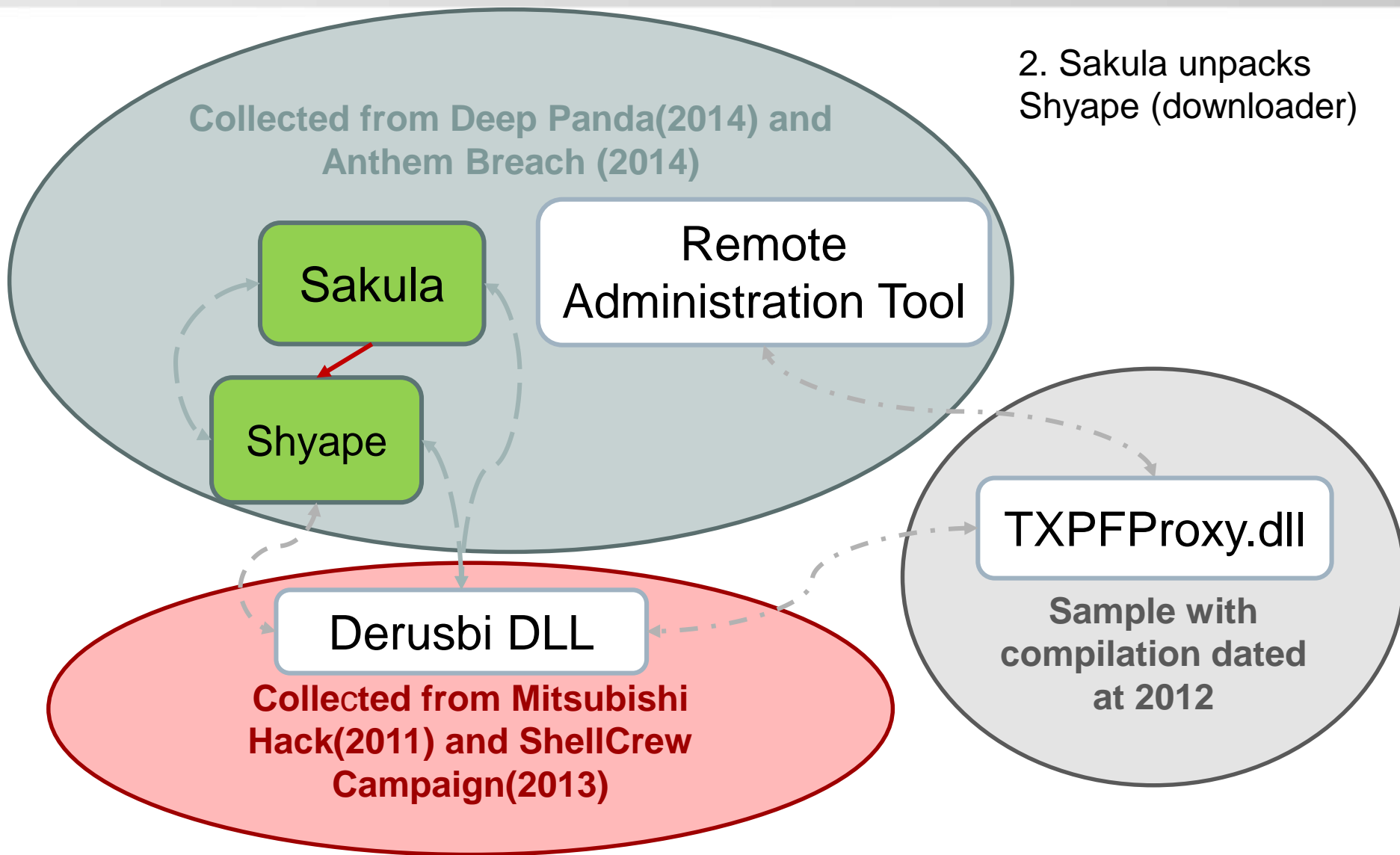


- Timeline
- 2008 – Earliest sample with compile time Aug 3, 2008
 - » (md5: 338e4deb0be7769ef2c9d7080fb56154)
- 2011 – Mitsubishi Heavy Industries hack (discovered Oct, 2011)
 - » (md5: 1cd7835b9ac253a72f8cd94405100d62) (Ref: [ixoxiブログ](#))(compile time Apr 15,2011)
- 2014 – CareFirst BlueCross BlueShield hack (by the work of Sakula)
 - » Revealed In May 2015
 - » 1.1 millions customer information breached
 - » Actual took place at June 2014 (Ref: [CareFirstAnswers](#))
- 2015 – Anthem hack (by the work of Sakula)
 - » Revealed in Mar 2015
 - » 78.8 million people information breached (Ref : [AnthemFacts](#))
 - » Data is stolen around Dec 2014 (Ref: [AnthemFacts](#))
 - » Part of the Deep Panda Campaign

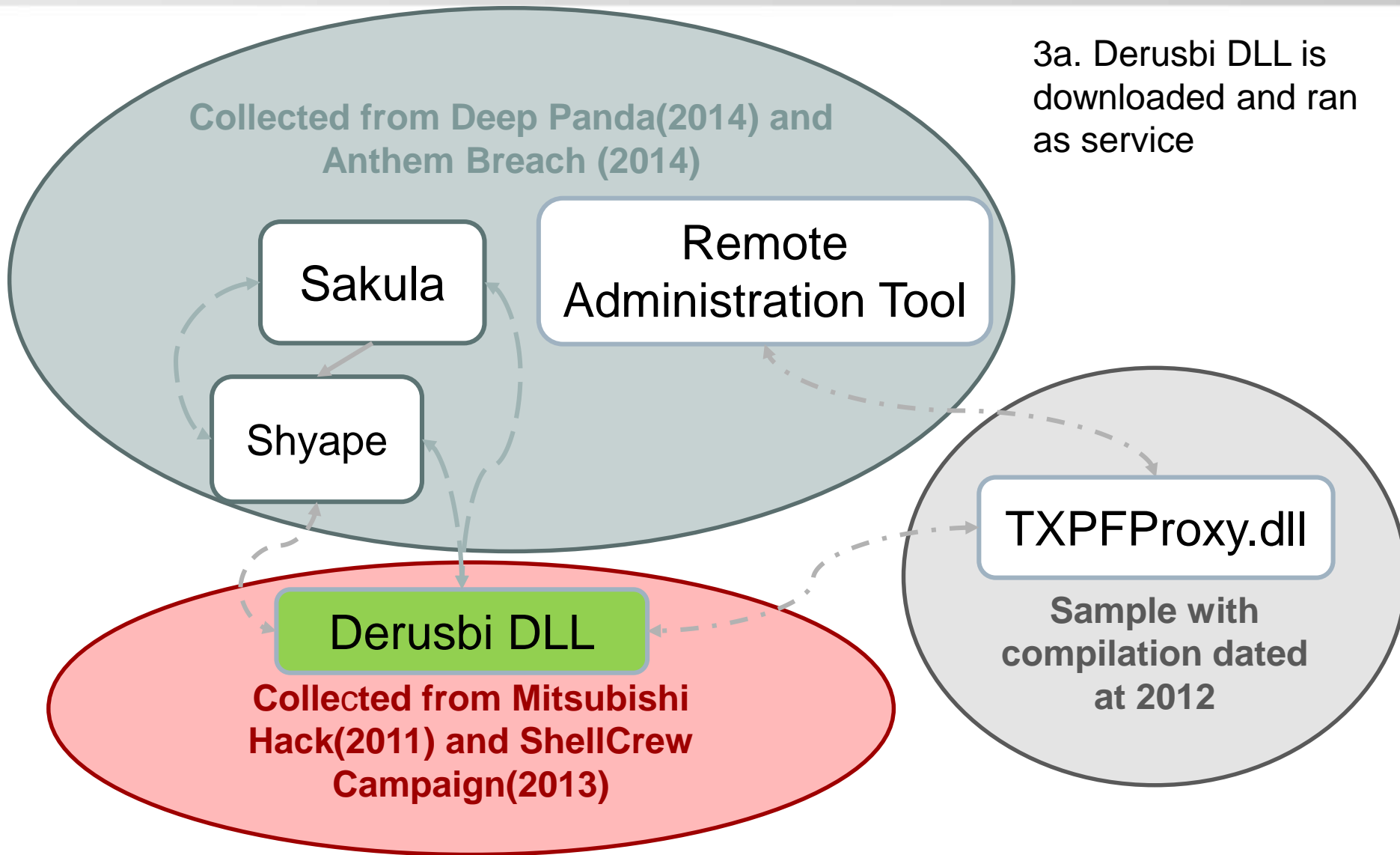
Possible Infection Routine



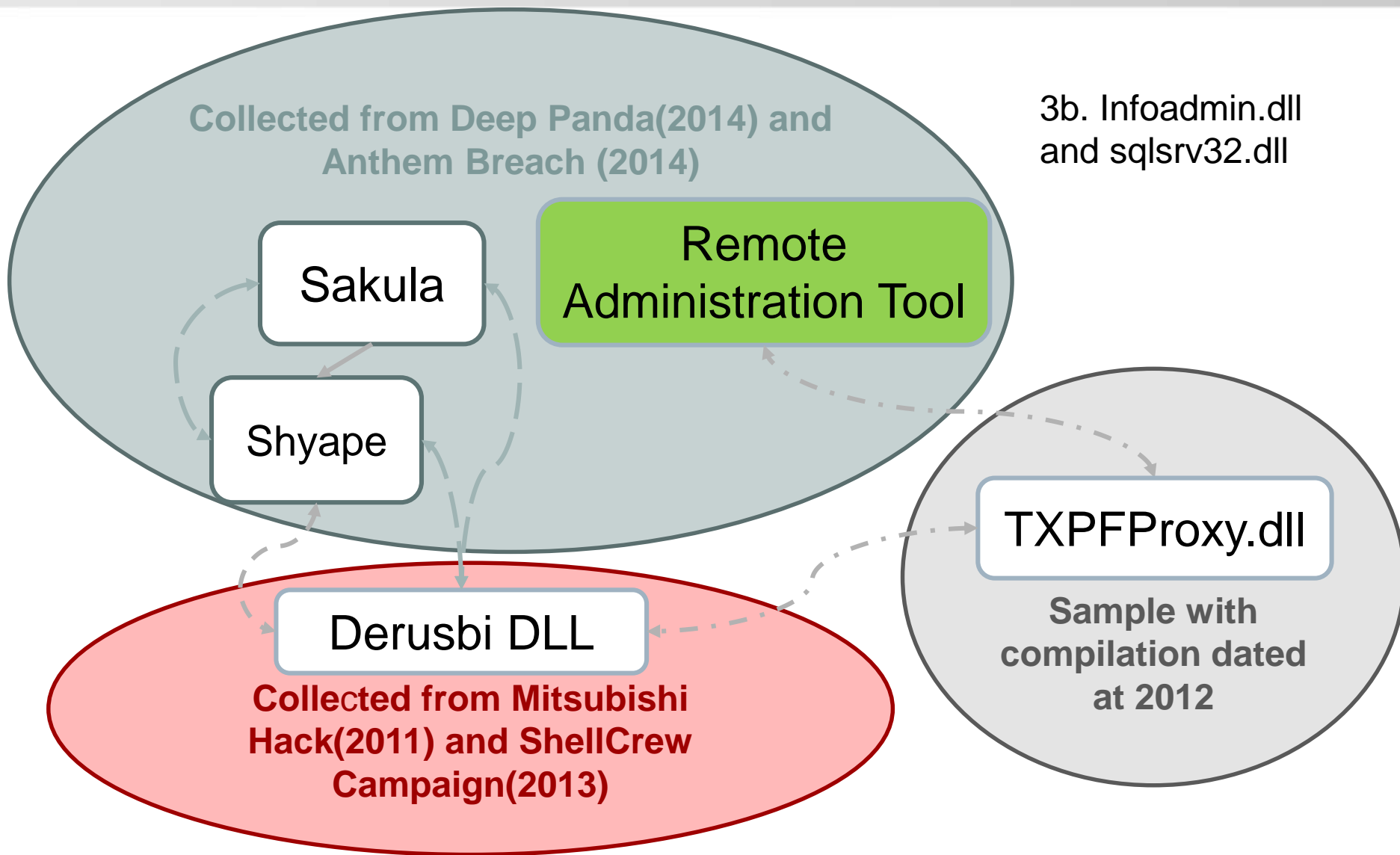
Possible Infection Routine



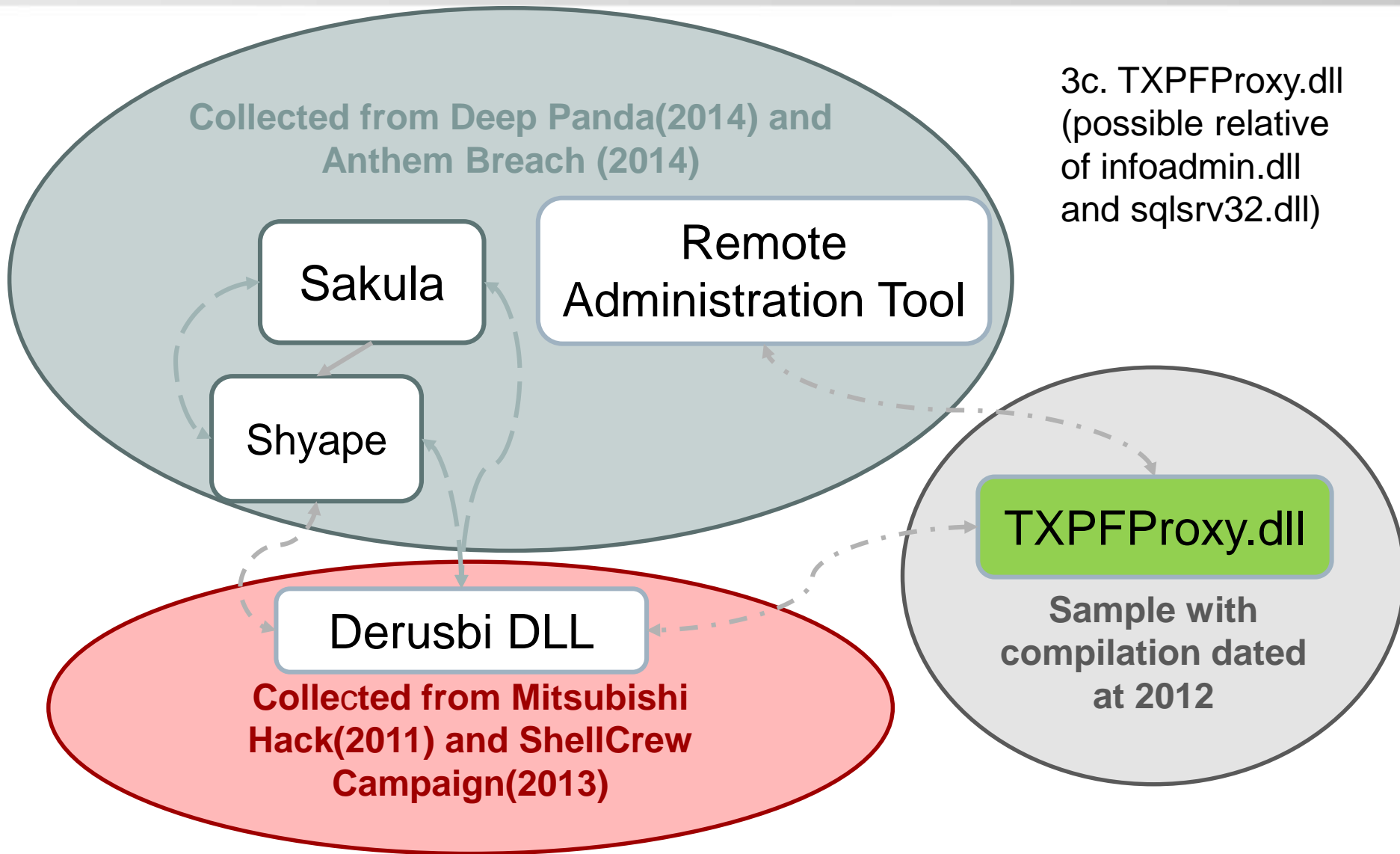
Possible Infection Routine



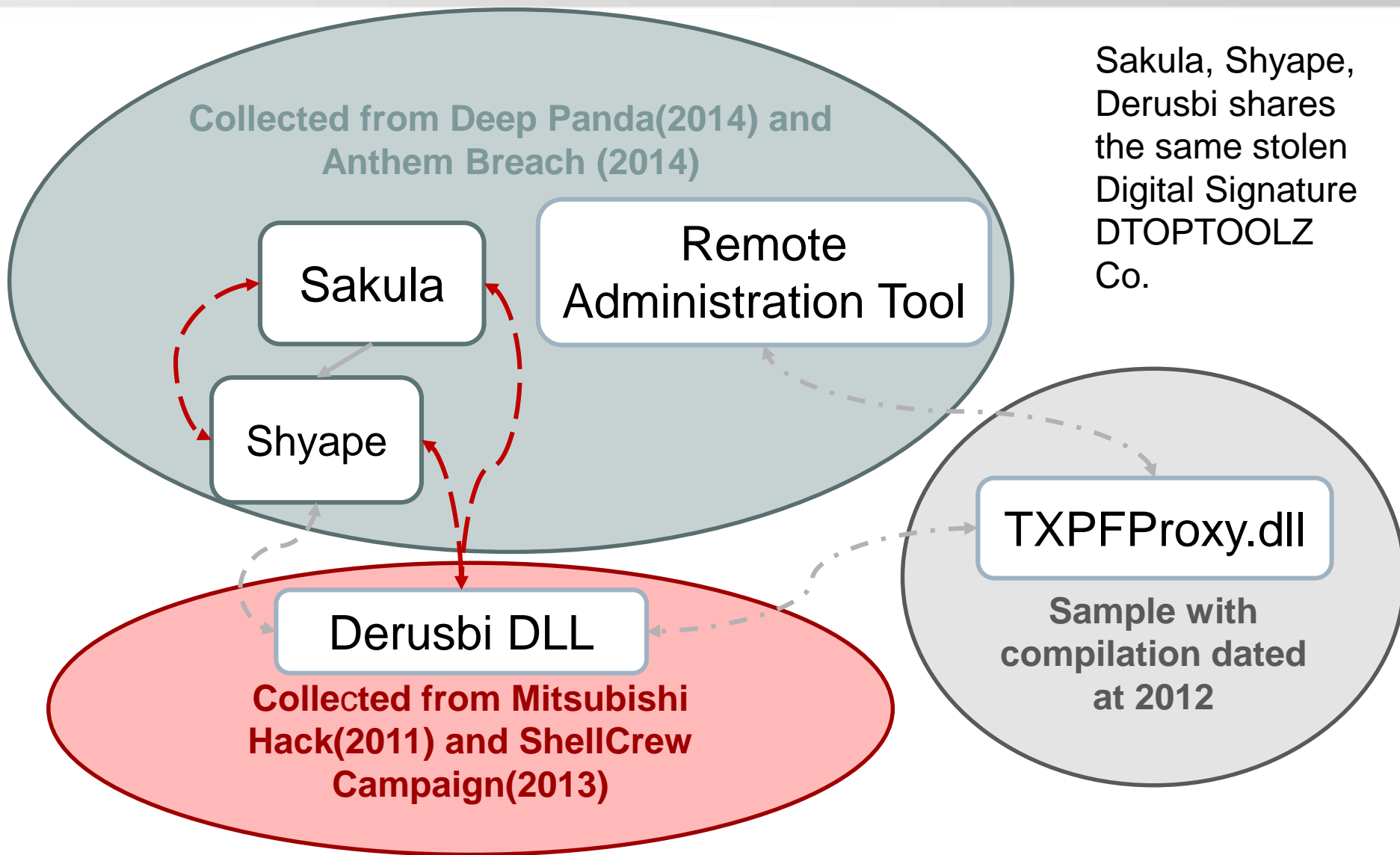
Possible Infection Routine



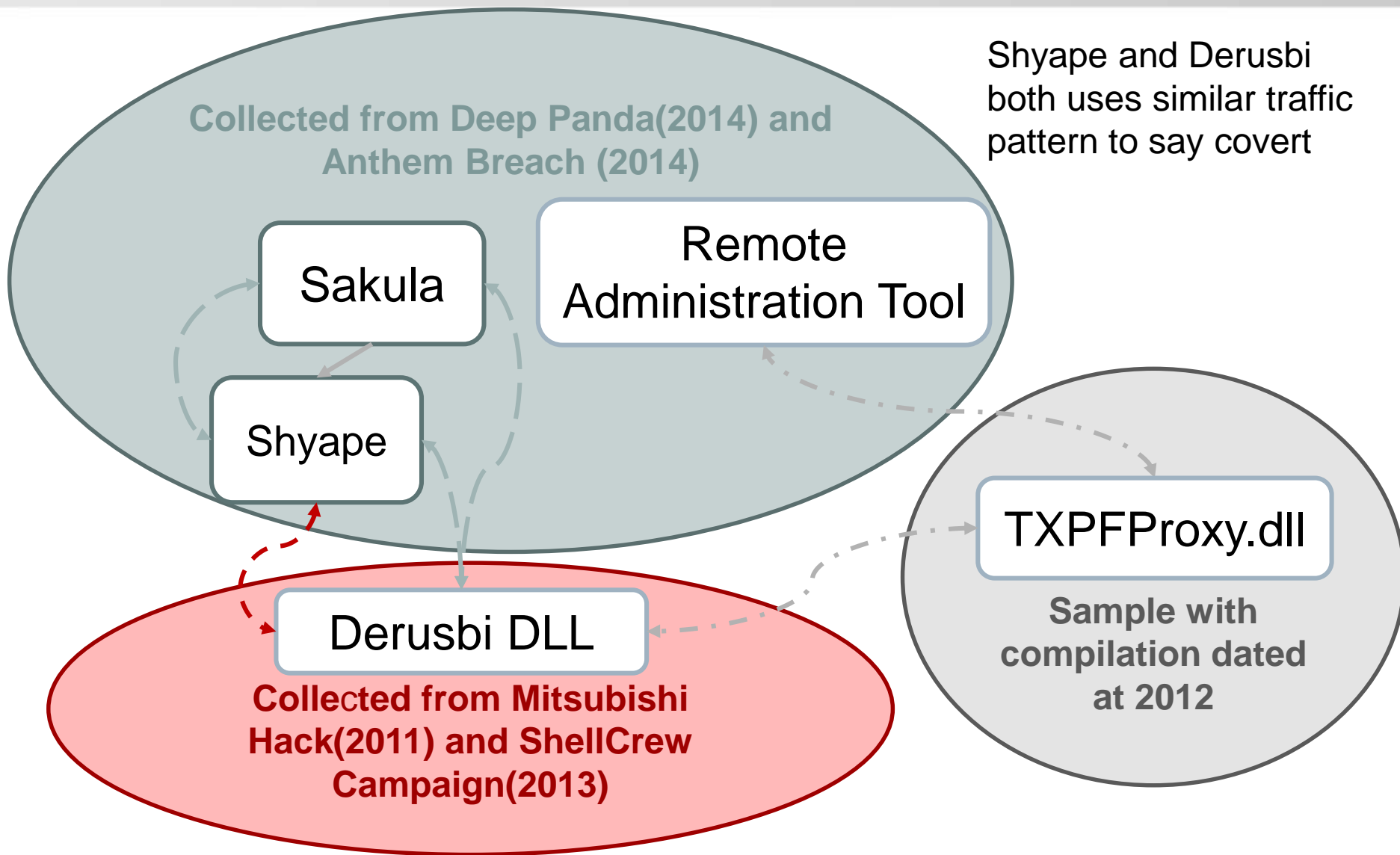
Possible Infection Routine



Similarities

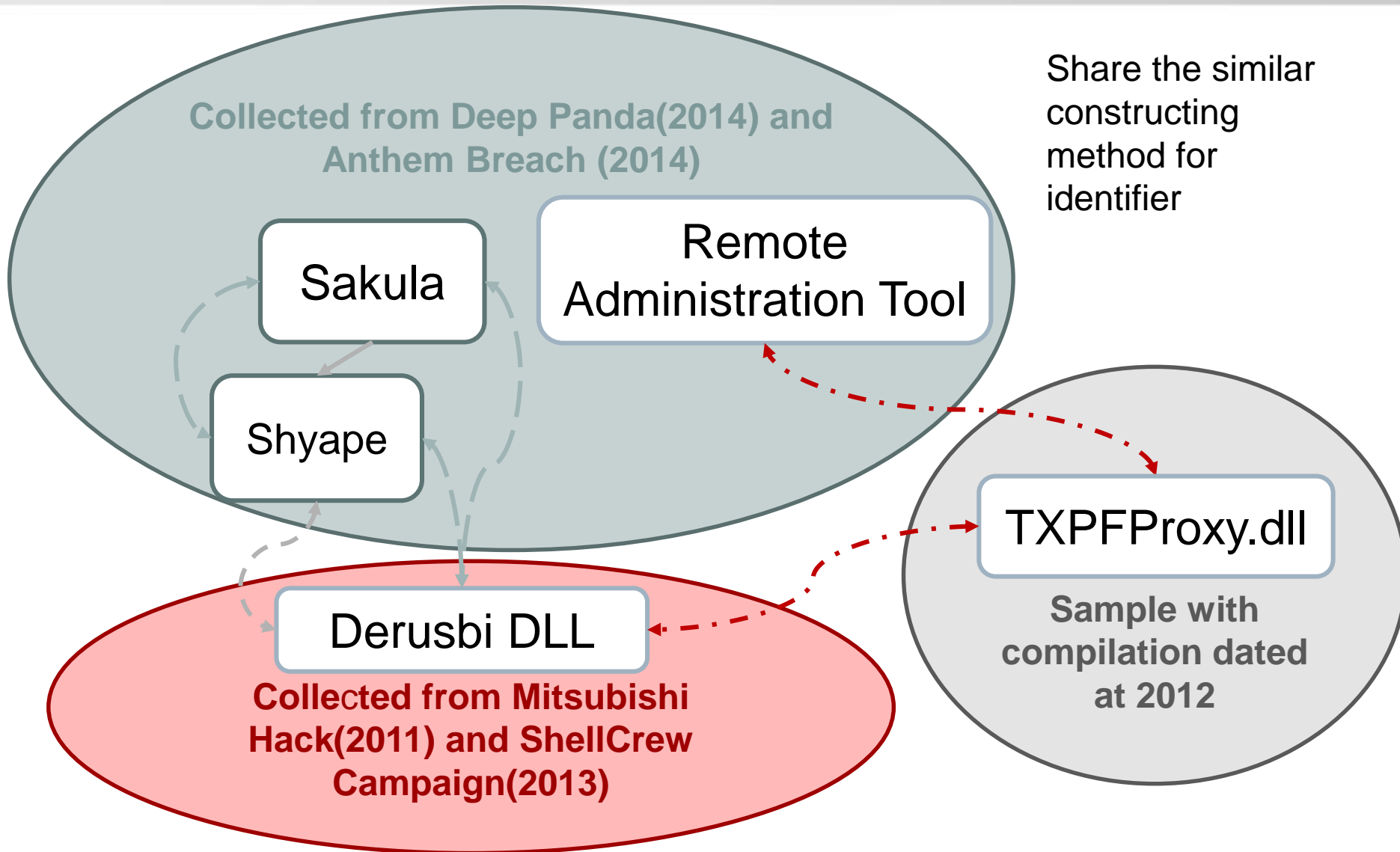


Similarities



```
aGetPhotosQue_0 db 'GET /Photos/Query.cgi?loginid=%d HTTP/1.1' ; DATA XREF: SER
db 'User-Agent: Mozilla/4.0 (compatible;
db 'Host: %s:%d', 0Dh, 0Ah
db 'Cache-Control: no-cache', 0Dh, 0Ah
db 'Pragma: no-cache', 0Dh, 0Ah
db 'Connection: Keep-Alive', 0Dh, 0Ah
db 0Dh, 0Ah, 0
```

Similarities



Similarities

```

loc_10006680:
mov     edx, [ebp+VersionInformation.dwBuildNumber]
push    edx
mov     eax, [ebp+var_8]
and     eax, 0FFFFFFh
push    eax
lea     ecx, [ebp+Dest]
push    ecx
push    offset a$SP$d (Build %d)
mov     edx, [ebp+Dest]
add     edx, 60h
push    edx
call    wsprintfA
add     esp, 14h
mov     [ebp+Buffer.dwLength], 20h
lea     eax, [ebp+Buffer]
push    eax
call    ds:GlobalMemoryStatus
mov     ecx, [ebp+Buffer.dwTotalPhys]
shr     ecx, 80h
shr     ecx, 80h
add     ecx, 1
push    ecx
push    offset a0mb : "%dMB"
mov     edx, [ebp+Dest]
add     edx, 40h
push    edx
call    wsprintfA
add     esp, 0Ch
mov     eax, 1

```

TXPFProxy.dll

```

loc_10005CFC:
movzx   eax, word ptr [ebp+VersionInformation.dwBuildNumber]
push    eax
lea     eax, [ebp+VersionInformation.szCSDVersion+80h]
push    eax
lea     eax, [ebp+var_18]
push    offset a$P$d : "%s (%d)"
push    eax
call    ds:sprintf
add     esp, 10h
jmp     loc_10005DEC

```

Remote Administration Tool (sqlsrv32.dll)

```

loc_10004676:
mov     ecx, [esp+15Ch+var_0]
mov     esi, ds:usprintfA
and     ecx, 0FFFFFFh
lea     edx, [esp+15Ch+var_13C]
push    ecx
push    edx
lea     eax, [ebx+800h]
push    offset a$SP$d : "%s SP%d"
push    eax
call    esi ; usprintfA
add     esp, 10h
push    offset aKernel32_dll_1 : "Kernel32.dll"
call    ds:LoadLibraryA
test     eax, eax
jnz     short loc_1000468A

loc_1000468A:
pop     edi
pop     esi
pop     ebp
pop     ebx
add     esp, 14Ch
retn

loc_10004680:
push    offset aGlobalMemoryStatusEx
push    eax
call    ds:GetProcAddress
test     eax, eax
jz      short loc_10004700

loc_10004700:
lea     ecx, [esp+15Ch+var_11C]
mov     [esp+15Ch+var_11C.dwLength], 40h
push    ecx
call    eax
mov     eax, [esp+15Ch+var_11C.dwTotalPhys]
mov     edx, [esp+15Ch+var_11C.dwAvailPhys]
mov     ecx, 14h
call    __aui shr
push    edx
push    eax
lea     edx, [ebx+800h]
push    offset a0mb : "%dMB"
push    edx
call    esi ; wsprintfA
add     esp, 10h
jmp     short loc_1000472D

loc_10004700:
lea     eax, [esp+15Ch+var_11C]
mov     [esp+15Ch+var_11C.dwLength], 20h
push    eax
call    ds:GlobalMemoryStatus
mov     ecx, [esp+15Ch+var_11C.dwTotalPhys]
lea     edx, [ebx+800h]
shr     ecx, 14h
inc     ecx
push    ecx
push    offset a0mb : "%dMB"
push    edx
call    esi ; wsprintfA
add     esp, 0Ch

```

Derusbi Collected from ShellCrew Campaign

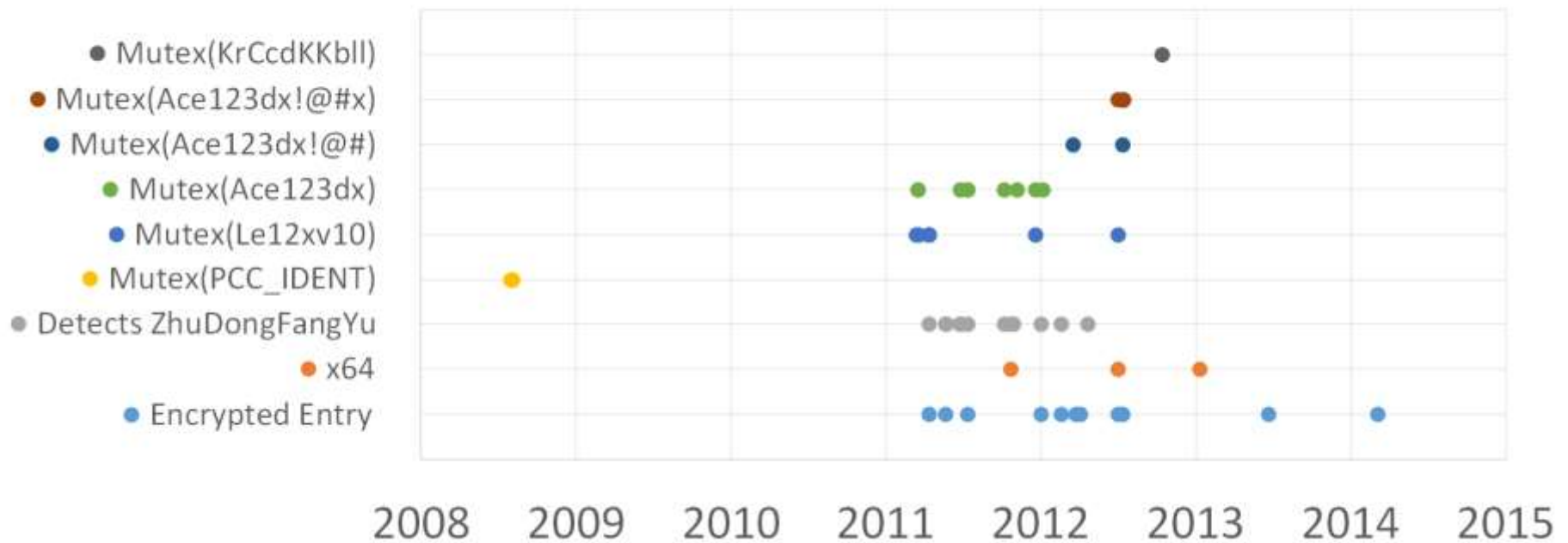
Variants of Derusbi



Variants of Derusbi



Variant Characteristics Against Compilation Time



Variants of Derusbi



Variant Characteristics Against Compilation Time



Variants of Derusbi



Variant Characteristics Against Compilation Time



Variants of Derusbi



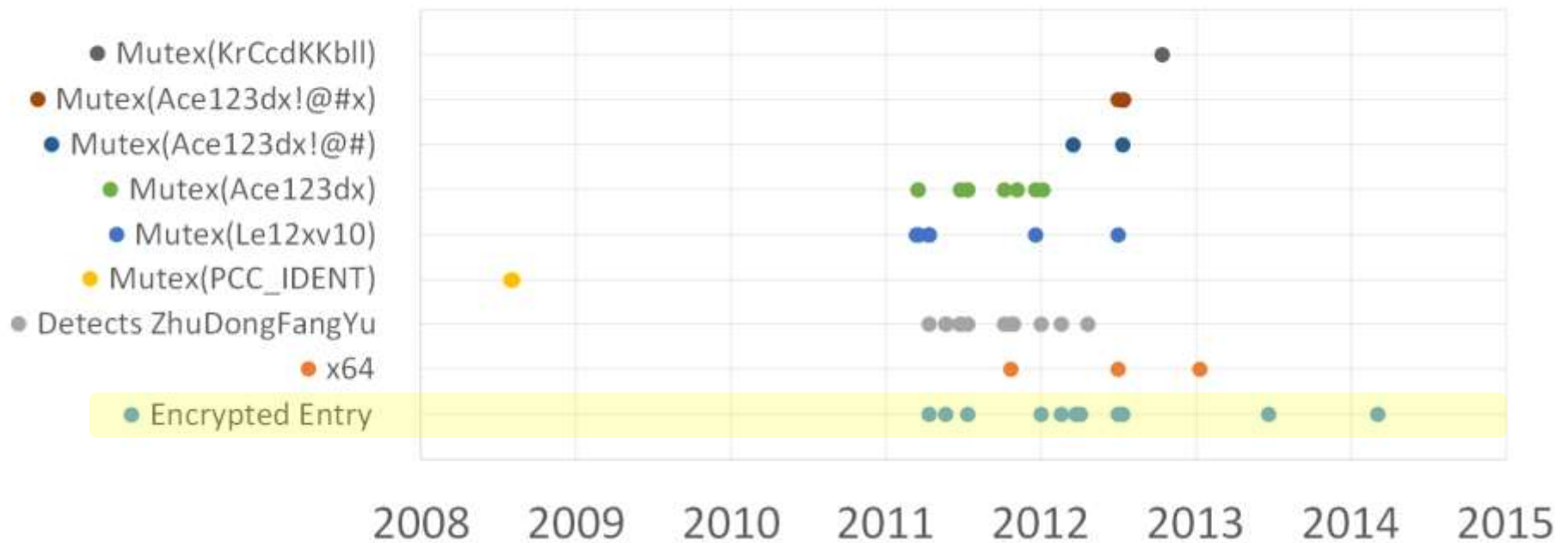
Variant Characteristics Against Compilation Time



Variants of Derusbi



Variant Characteristics Against Compilation Time





- Some notes:
 - » 64-bit version first seen in 2011 – somewhat rare
 - » Newer samples don't necessarily use the newest version of a specific class
 - » Much more features in samples from 2013/2014 versus 2008

Technical Analysis





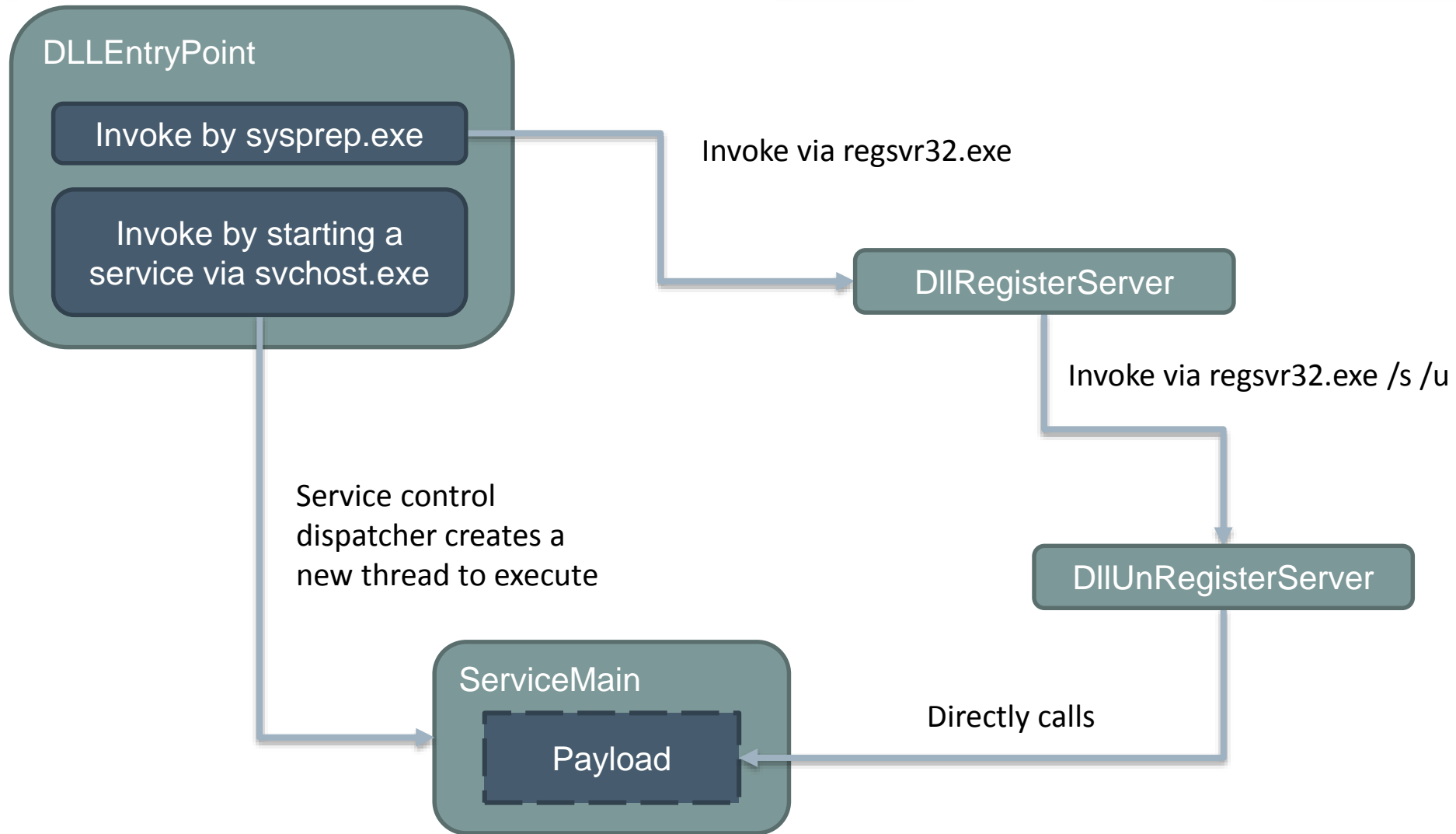
- DllEntryPoint
 - » Initialization
 - » Calls regsvr32.exe
 - » If sample is packed, unpack the export functions
- DllRegisterServer
 - » Persistence Management
- DllUnregisterServer
 - » Invoke Payload/BDSocket Thread
- ServiceMain
 - » Main code
 - » Contains the Payload/BDSocket Thread

Technical Analysis

Persistence Management



Derusbi Loading Sequence





- Decrypt and store built-in configuration at
 - » Key: HK_Local_Machine\Software\Microsoft\RPC
 - » Subkey: Security
 - » Data: xor(not(one-byte key))[Decrypted Configuration]
- Backup the current file to %SystemFolder% with filename
 - » [hardcoded-prefix]{randomstring}.[hardcoded-extension]
- Store the persistent DLL path in
 - » Key: HK_LOCAL_MACHINE\System\CurrentControlSet\Service\
{Persistent Service Name}\Parameter
 - » Subkey: ServiceDLL

Built-in Configuration



Address	Hex dump	ASCII
10020D50	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020D60	00 00 00 00 00 00 00 00 0A 00 00 00 68 65 6C 70	
10020D70	73 76 63 00 00 00 00 00 00 00 00 00 00 00 00 00	svc help
10020D80	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020D90	31 37 32 2E 32 32 2E 31 36 31 2E 31 39 3A 38 30	172.22.161.19:80
10020DA0	38 30 00 00 00 00 00 00 00 00 00 00 00 00 00 00	80
10020DB0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020DC0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020DD0	63 3A 5C 77 69 6E 64 6F 77 73 5C 73 79 73 74 65	c:\windows\system
10020DE0	60 33 32 5C 00 00 00 00 00 00 00 00 00 00 00 00	m32\
10020DF0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020E00	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020E10	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	
10020E20	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	

Persistent service name

Beacon URL

File path where the Derusbi client is stored on the computer under a different name

Built-in Configuration



Address	Hex dump																ASCII
10022601	00	00	00	00	00	00	00	54	7C	26	63	66	51	33	55		T!&cfQ3U
10022611	55	4F	38	5E	34	3B	56	5A	7C	58	6D	2D	7E	4A	6C	2B	U08^4;U2!Xm~"Jl+
10022621	34	34	77	45	54	58	5E	3E	6A	31	65	31	4B	2B	4A	69	44wETX^>j1e1K+Ji
10022631	5F	59	68	66	57	3E	31	51	5C	26	37	76	5D	25	77	67	_YhfW>1Q\&7v]Xwg
10022641	23	55	6E	41	63	3D	77	32	30	32	2E	38	36	2E	31	39	#UnAc=w202.86.19
10022651	30	2E	33	3A	38	30	00	35	2E	79	44	2A	70	41	6B	3A	0.3:80 5.yD*pAk:
10022661	5D	27	6C	3A	2D	69	64	28	4A	4B	75	60	54	73	3D	2C	J'l:-id(JKu'Ts=,
10022671	4D	6B	48	74	71	55	2A	51	67	78	53	42	2B	35	66	3A	MkHtqU*QgxSB+5f:
10022681	24	77	4F	68	5B	40	26	6C	4C	79	62	4F	49	63	5B	67	\$wOh[0&llYb0Ic[g
10022691	43	30	7C	6D	63	6D	75	57	7A	51	32	6F	69	5B	40	59	C0!mcnuWzQ2oi[0Y
100226A1	5B	59	46	70	36	5B	29	73	45	65	21	55	70	32	29	53	[YFp6[]sEe*Up2)S
100226B1	57	22	67	36	47	4B	71	4E	5C	43	7E	76	76	68	64	59	W"geGHqN\C"vvhdy
100226C1	47	4F	67	31	28	41	3B	30	70	21	4D	25	62	2B	35	5E	G0g1(A;0p!M%b+5^
100226D1	44	7C	50	61	78	4F	59	60	29	51	5D	59	46	3D	37	32	D!PaxOY')Q]YF=72
100226E1	61	56	2D	7E	54	64	7C	36	4E	58	63	70	24	6E	35	2E	aU~"Td!6NXcp\$5.
100226F1	3D	37	5C	5C	2A	74	50	5C	6E	6C	4B	2E	40	5B	5F	5D	=7\~*tP~n lK.@[_]
10022701	68	79	2D	7C	21	77	25	5B	37	70	51	7B	60	61	6C	5C	hy~!tW%[7p0t~a\
10022711	40	49	6C	3D	59	6F	3B	55	60	65	79	27	70	34	5A	42	@Il=Yo;U'ey~p42B
10022721	51	5A	4D	6B	76	4F	41	47	5D	62	21	61	73	65	5A	3E	Q2MkvOAG]b~aseZ>
10022731	6A	50	37	33	60	73	36	26	29	57	71	37	66	3D	4B	3A	jP7S~x6&)Wq?f=H:
10022741	27	42	4E	78	6D	3D	27	14	00	00	00	77	75	61	75	73	*BNxm~'q wuauS
10022751	65	72	76	00	48	25	7B	74	27	5E	7E	79	50	7D	74	4D	erU HZ(t'~yP)tM
10022761	30	35	5B	67	6C	75	60	34	40	56	4F	00	00	00	00	00	05[glu~40UO
10022771	74	7E	6E	50	2A	44	4B	63	69	69	31	5E	7D	54	72	7C	t~nF*DHcii1^>Tr!
10022781	79	57	64	36	68	65	7E	70	57	27	35	67	43	41	25	00	yWd6he~pW~5gCA%
10022791	31	72	26	24	6B	55	5F	26	4D	2E	46	53	34	2A	78	00	1r&\$kU_&M.FS4*x
100227A1	36	75	5D	48	38	67	40	7D	3A	70	31	4B	25	44	7C	00	6u]H8g0):p1KXD!
100227B1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
100227C1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
100227D1	44	3B	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00;
100227E1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
100227F1	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
10022801	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
10022811	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	



- If McAfee's anti-virus service is detected, it would not use regsvr32.exe to invoke the DllUnregisterServer export function
- It will copy of regsvr32.exe to update.exe, run update.exe and then invoke the DllUnregisterServer export function

Persistence Management – Registry Setup

Key: *HK_LM\Software\Microsoft\RPC*
Sub Key: Security

xor(not(one-byte key))[Decrypted Configuration]

Identifier

Persistent Service Name

```
10020D50|00 00 00 00|00 00 00 00|00 00 00 00|00 00 00 00|
10020D60|00 00 00 00|00 00 00 00|0A 00 00 00|68 65 6C 70|
10020D70|73 76 63 00|00 00 00 00|00 00 00 00|00 00 00 00|
10020D80|00 00 00 00|00 00 00 00|00 00 00 00|00 00 00 00|
```

svc help

Key: *HK_LM\Software\Microsoft\Windows NT\Current Version\Svchost*
Sub Key: netsvcs

Service Name

Persistent Service Name

Service Name

Service Name

Key: *HK_LM\System\CurrentControlSet\Service\Persistent Service Name\Parameter*
Sub Key: ServiceDll

Path to Derusbi DLL at %systemRoot%

Technical Analysis

Payload



■ Main Thread



- SeDebugPrivilege
- SeLoadDriverPrivilege
- SeShutdownPrivilege
- SeTcbPrivilege

Optional Embedded Driver

- Main Thread



- Not all samples contain an embedded driver
- XOR-encrypted, with 4-byte key
- Conditions for decrypting and loading driver
 - » 360's ZhuDongFangYu.exe must not be running (optional)
 - » The username of the current process must be "system"



■ Main Thread



■ Example Drivers:

- » Keylogger
- » USB/Disk infector
- » Network hooking driver



- Derusbi Sample (MD5: 92d18d1ca7e66539873be7f5366b04d1)
- Iterate all directories on the disk
- Drop Derusbi when service DLLs found
- Create autorun.inf to auto-register Derusbi when the infected drive is connected to a computer

■ Main Thread



■ Second Thread



Technical Analysis

Built-in modules



Built-in Classes

- Written in C++
- RTTI information!
 - » Thanks to IDA [ClassInformer](#) plugin

```
.rdata:10017EBC      ; class PCC_SYS: PCC_BASEMOD;  (#classinformer)
.rdata:10017EBC 0C 94 01 10      dd offset ??_R4PCC_SYS@@@6B@ ; const PCC_SYS::`RTTI Complete Object Locator'
.rdata:10017EC0      ; const PCC_SYS::`vftable'
.rdata:10017EC0 89 D1 00 10      ??_7PCC_SYS@@@6B@ dd offset PCC_SYS_ctor ; DATA XREF: sub_1000D1A9+3↑o
.rdata:10017EC0      ; BD_THREAD_main+271↑o
.rdata:10017EC4      dd offset return1
.rdata:10017EC8 DD D1 00 10      dd offset PCC_SYS_CLEANUP
.rdata:10017EC8 CA D2 00 10      dd offset PCC_SYS_PROC_PACKET
.rdata:10017ED0 10 D2 00 10      dd offset PCC_SYS_READ_WAITING_DATA
.rdata:10017ED4 0C 38 00 10      dd offset malloc_0
.rdata:10017ED8 36 38 00 10      dd offset free_0
```





- Unfortunately, some 2014 samples uses updated classes

Some Built-in Class Names

- INTERNAL_CMD
- PCC_BASEMOD
- PCC_CMD
- PCC_FILE
- PCC_MISC
- PCC_PROXY
- PCC_SYS

Built-in Class Hierarchy

- All command classes are child classes of abstract class PCC_BASEMOD

Vftable	Methods	Flags	Type	Hierarchy
 10016B34 7			PCC_BASEMOD	PCC_BASEMOD:
 10016B54 1			PCC_CMD	PCC_CMD:
 10016BE4 7			PCC_FILE	PCC_FILE: PCC_BASEMOD;
 1001725C 7			PCC_MISC	PCC_MISC: PCC_BASEMOD;

Built-in Command Class Functions

■ PCC_BASEMOD

```
.rdata:10016B30 ; class PCC_BASEMOD: (#classinformer)
.rdata:10016B30 dd offset ??_R4PCC_BASEMOD@@@6B@ ; const PCC_BASEMOD::`RTTI Complete Object Locator'
.rdata:10016B34 ; const PCC_BASEMOD::`vftable'
.rdata:10016B34 ??_7PCC_BASEMOD@@@6B@ dd offset PCC_BASEMOD_dtor
.rdata:10016B34 ; DATA XREF: sub_1001524B:loc_10003802↑to
.rdata:10016B34 ; PCC_BASEMOD_dtor+A↑to ...
.rdata:10016B38 dd offset return1
.rdata:10016B3C dd offset _purecall
.rdata:10016B40 dd offset _purecall
.rdata:10016B44 dd offset _purecall
.rdata:10016B48 dd offset malloc_0
.rdata:10016B4C dd offset free_0
```

■ INTERNAL_CMD

```
.rdata:10016B58 ; class INTERNAL_CMD: PCC_BASEMOD; (#classinformer)
.rdata:10016B58 dd offset ??_R4INTERNAL_CMD@@@6B@ ; const INTERNAL_CMD::`RTTI Complete Object Locator'
.rdata:10016B5C ; const INTERNAL_CMD::`vftable'
.rdata:10016B5C ??_7INTERNAL_CMD@@@6B@ dd offset InternalCmd_dtor
.rdata:10016B5C ; DATA XREF: init_INTERNAL_CMD+10↑to
.rdata:10016B5C ; INTERNAL_CMD_init+11↑to
.rdata:10016B60 dd offset return1
.rdata:10016B64 dd offset INTERNAL_CMD_CLEANUP
.rdata:10016B68 dd offset INTERNAL_CMD_PROC_PACKET
.rdata:10016B6C dd offset INTERNAL_CMD_READ_WAITING_DATA
.rdata:10016B70 dd offset malloc_0
.rdata:10016B74 dd offset free_0
.rdata:10016B78 dd offset INTERNAL_CMD_WORK
```

- [Novetta, 2014](#) describes some of these functions for an older Derusbi sample



- There is also a default handler
 - » packet_type/class_id: 100h
- Some of its functions:
 - » Terminate current connection (deprecated)
 - » Cleanup data stored in the different modules
 - » Backup configuration to registry, set current file to be deleted on reboot, terminate current process immediately
 - » Terminate after current jobs
 - » Install a new DLL



- INTERNAL_CMD (supersedes PCC_CMD class)
 - » 2011 – Present
 - Some samples from 2012 do not have this class though
 - » Class ID: 5
 - » Interactive shell commands
 - » Has help/? functions!!!
 - » Common OS operations (v1.1)
 - cd, dir, md, rd, del, copy, ren, type, start
 - » Additional commands in v1.2
 - runas
 - reboot [-f]
 - shutdown [-f]
 - clearlog
 - wget [httpurl]



■ PCC_MISC

- » 2011 – Present
- » Most samples have this class
- » Class ID: 10
- » Mixture of numerical and text commands
- » Command IDs:
 - ID=1: save attached file to temp dir and load as DLL. Can remember up to 16 files.
 - ID=2: delete temp file. Attached filename must correspond to one of the 16 saved from command ID 1



■ PCC_MISC

- » 2011 – Present
- » Most samples have this class
- » Class ID: 10
- » Mixture of numerical and text commands
- » Text commands:
 - “pstore”: steals password information from IE and firefox and send to C2
 - “keylog”: send keylog info to C2
 - “info”: gathers system information and send to C2
 - » OS name and build number
 - » Network adapter info
 - » IE version
 - » Proxy server info
 - » AV info (Norton, 360, Kaspersky, Trend Micro, ESET, Avira)



■ PCC_SYS

- » 2008 – Present
- » Almost all samples have this class
- » Class ID: 4 (80h in older samples)
- » 4 types of numerical commands
 - Processes-related: enumerate and kill processes
 - Services-related: enumerate, start, stop, delete services
 - Registry-related: enumerate, create/delete keys, set/delete/replace values
 - Screenshot command
- » Each type contains its own command IDs



- PCC_FILE
 - » 2008 – Present
 - » Almost all samples have this class
 - » Class ID: 8 (84h in older samples)
 - » Numerical commands
 - Cleanup
 - Enumerate all drives
 - Find/rename/delete/copy/move file
 - Save a file to system
 - Recursively enumerate directory
 - Start new process
 - Recursively enumerate all drives

Current generation (2014 – Present)



- Old code, just packed
 - » Class structure and functions from 2011/2012
 - » Compatibility/on-going attack?

- New version
 - » Same payload delivery
 - » Updated built-in classes

- Still written in C++
- No RTTI information
- Updated/rewritten classes
 - » Custom code for creating new() objects
 - » New is_this_data_for_me() virtual function
 - » Dynamically decrypt embedded helper DLL during class initialization
 - Inject helper DLL into explorer.exe in class command handler function
 - Communicate with helper DLL using pipes
 - » Removed duplicate functionality in modules

Updated Built-in Classes – Con't



- Command IDs changed
- No more verbose commands
- No interactive shell
- PCC_SYS, PCC_FILE, default_handler functionality still there
- Identify newer OS like Win8 (but no Win 8.1 or 10)
- Processor architecture detection(x86, x64, IA64, ARM)

Conclusion



- Samples circulating between vendors
 - » Limited number of samples
 - » Delayed discovery
 - » Corrupt files
- To improve detection
 - » Class/modular structure
 - » IPS
 - » Sakula/Shyape

Summary



- Modular
- Fully-featured for stealth and espionage
- Targeted attacks
- Operations could take up to 2 years

Any questions?

{mpun, ericleung, ntan}@fortinet.com

FORTINET®