Microsoft

Using an expert system to provide automated malware analysis for non-experts…

Hermineh Tchagatzbanian
Senior Program Manager - MMPC
Microsoft

Microsoft

…or using machines to provide meaningful analysis for humans.

Heather Goudey
Content Publishing Manager - MMPC
Microsoft

MMPC: Our impact

Protection points:

MSE SCEP Intune MSRT Defender Hotmail Exchange Azure

Daily

150K samples
15M telemetry
45M scans
12 sig sets
3 sig releases

Our partners

Security industry + Trustworthy Computing,
Microsoft Digital Crimes Unit, Bing,
IE Smartscreen, Store

bing

Internet Explorer

Windows Phone

MICROSOFT DIGITAL CRIMES UNIT

Agenda

Introduction What's the problem?	Background What can we do?	The expert system How does it work?	Examples The result.

Introduction

Providing a complete solution

Detection and removal aren’t always enough

A malicious compromise without effective remediation beyond removing the malware in question can have negative and far-reaching consequences for affected users

The problems…

Analyzing malware is expensive

Modern malware is obfuscated, multi-component and complex

An accurate analysis can take days, if not weeks

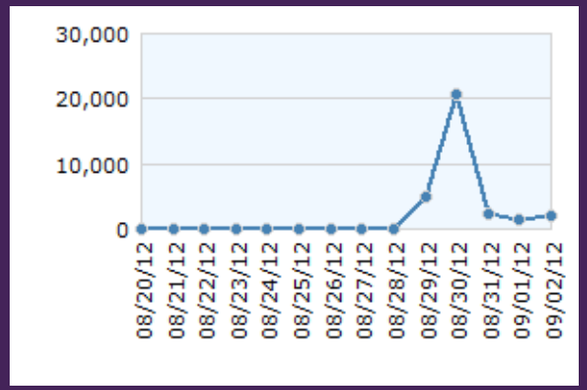
The cost of performing analysis for publishing is adding detections (*that* old chestnut)

There's a lot of it about

The sheer volume and distribution of malware in the wild makes describing it using manual analysis nigh-impossible

…we're trying…

Malware may be fleeting



Malware often has a short lifespan

Malware information has a time-limited utility – it needs to be there when users look for it

Late description = wasted effort and limited effectiveness

…to solve.

It changes

The behavior of malware families changes over time

Analysis must be ongoing and continuously updated to stay accurate

It’s global

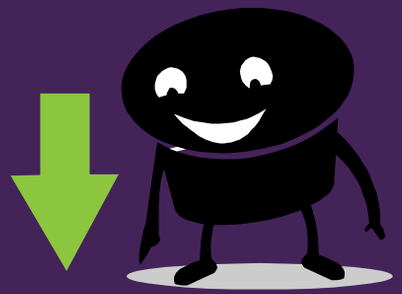
It’s not limited to a particular location

It’s not limited to particular populations

It’s not an ‘English’ problem

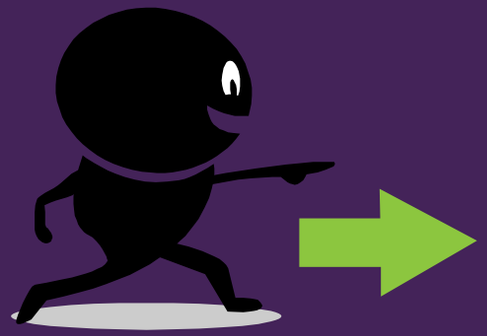
The manual description process

1. Reverse compiled code
2. Run the malware in a controlled environment
3. Closely monitor system changes and observe behavior



4. Add knowledge of what the behaviors mean and intelligence on context

Publishing the analysis



1. Analysts' notes are edited and formatted as appropriate for the audience
2. Content added on additional remediation beyond the product
3. The description is published

Background

Current approaches

jar-gon *noun*

Definition

1.
   a) Confused unintelligible language
   b) A strange, outlandish, or barbarous language or dialect
   c) A hybrid language or dialect simplified in vocabulary and grammar and used for communication between peoples of different speech
2. The technical terminology or characteristic idiom of a special activity or group
3. Obscure and often pretentious language marked by circumlocutions and long words

http://www.merriam-webster.com/dictionary/jargon

Malware is a human problem

What's happened?

How did the malware get on my computer?

What did it do to my computer?

What does it mean?

What do the malware's changes mean to me personally?
What was the malware's effect on the confidentiality, integrity and availability of my computer resources?

What do I do now?

What steps to I have to take? (Including remediation of the possible 'human' problems that may have occurred as a consequence of the infection).

A good malware analysis…

Provides information an affected user can USE

Gets to the point and is purposeful

Understands the user's situation (*I'm infected!*) and provides what they need to know

Doesn't patronize - it reassures and speaks in the user's language

Avoids jargon

Uses common everyday words and phrases

Uses language that inspires confidence and provides encouragement

Is an opportunity to educate

Gives the user something extra to help them avoid further compromise

The expert system

Automatic malware descriptions

Mimic manual descriptions

Are driven by malware behaviour

Use small templates

Use a rule-based expert system

Description template example

<malware> changes the start page for Internet Explorer to <*StartPageURL*> by making the following registry modification:

Adds value: "*Start Page*"
With data: "<*StartPageURL*>"
To subkey: *HKCU\Software\Microsoft\Internet Explorer\Main*

Each template has

One or more rules associated with it

Sections & subsections

A prescribed order within each subsection

Text formatting rules

Types of rules

Rules are based on:

- Malware type
- Malware name
- Malware activities
- Behavioral knowledge
- Created file content
- Multiple runs of malware
- Combinations of rules

Example rule structure

```
If (regKey="HKCU\Software\Microsoft\Internet Explorer\Main
"  && Value="Startpage" && Data <>"")
Then
        store Data value and mark the Rule passed
```

Applied rule example

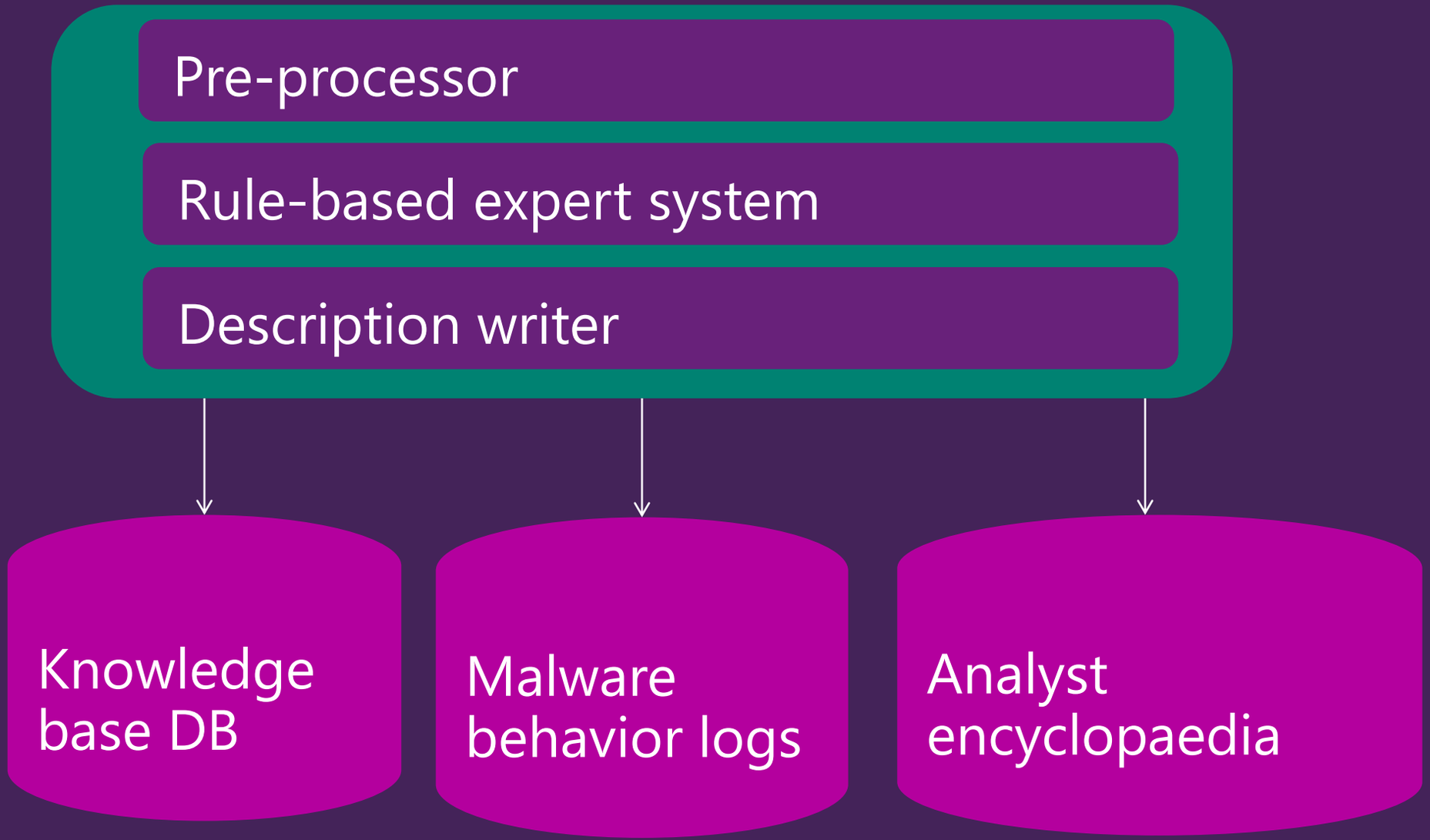
Payload

Modifies browser settings

The malware changes the start page for Internet Explorer to *http://m-.5-b-e-n-t-f-p-p-7-1-1-0-7-c-q-0-3-00-6-u-7-t-1-n-f-u-d-g-e.info* by making the following registry modification:

Adds value: "*Start Page*"
With data: "*http://m-.5-b-e-n-t-f-p-p-7-1-1-0-7-c-q-0-3-00-6-u-7-t-1-n-f-u-d-g-e.info*"
To subkey: *HKCU\Software\Microsoft\Internet Explorer\Main*

Automation modules



Examples

Example of a template

<malware name> ensures the worm copy is executed when certain Windows applications are run, including but not limited to<Clean Applications>. It does this by making the following registry modifications:

Adds value: "Debugger"
With data: "<Malware File name>"
To subkey: HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\<Product Name>

where < Product Name > may be any of the following
<repeat>
{
- <application name>

}

Example rules

Rule-1
IF (registry key =’HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\’ && Data= malware file name && and Value =’Debugger’)
THEN
Get from behavior log the subkey and Data and store in system database.
Rule-2
IF (Rule-1 is true)
THEN
Get from behavior log <Application name> and store in system database
Repeat Rule-1 and Rule-2 until all such subkeys are exhausted
Rule-3
IF(Rule-2 is true && Application names are known to our system)
THEN
Get from the system top 3 product names that map <Application name>

Example application of rules

Installation

The malware ensures the worm copy is executed when certain Windows applications are run, including but not limited to security products, Registry Editor and Task Manager. It does this by making the following registry modifications:

Adds value: "*Debugger*"
With data: "*c:\documents and settings\administrator\administrator1\winlogon.exe*"
To subkey: *HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\<product name>*

where <product name> may be any of the following:
_avp.exe
.
.
_taskmgr.exe
_regedt32.exe

An excerpt of the finished product

Technical Information (Analysis)

TrojanDownloader:Win32/Banload.AOQ is a member of Win32/Banload - a family of trojans that downloads other malware. Banload is usually used to download and install members of the Win32/Banker and Win32/Bancos families onto affected computers. Win32/Banker and Win32/Bancos are trojans that steal banking credentials and other sensitive data, and send it back to a remote attacker.

Installation

TrojanDownloader:Win32/Banload.AOQ creates the following files on an affected computer:

- *c:\documents and settings\administrator\local settings\application data\mdl.dat*
- *c:\documents and settings\administrator\local settings\application data\mg.dat*
- *c:\documents and settings\administrator\local settings\application data\msgs.cpl*
- *c:\documents and settings\administrator\local settings\application data\t9lixwg6.txt*
- *c:\documents and settings\administrator\local settings\application data\tpp.dat*
- *c:\documents and settings\administrator\local settings\temp\hcb7.bat*

Payload

Modifies system security settings
TrojanDownloader:Win32/Banload.AOQ disables the LUA (Least Privileged User Account), also known as the "administrator in Admin Approval Mode" user type, by making the following registry modification:

Adds value: *"EnableLUA"*
With data: *"0"*
To subkey: *HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System*

Note: Disabling the LUA allows all applications to run by default with all administrative privileges, without the user being prompted for explicit consent.

Contacts remote host
The malware may contact a remote host at *topshow2012.com* using port 80. Commonly, malware may contact a remote host for the following purposes:

- To report a new infection to its author
- To receive configuration or other data
- To download and execute arbitrary files (including updates or additional malware)
- To receive instruction from a remote attacker
- To upload data taken from the affected computer

This malware description was produced and published using our automated analysis system's examination of file SHA1 *2ad7b0f8d2d448b96eed1fb2daca0d09964089a0*.

http://www.microsoft.com/security/portal/Threat/Encyclopedia/Entry.aspx?Name=TrojanDownloader%3aWin32%2fBanload.AOQ

Conclusion

Our goal

To create a system that would publish malware descriptions that are:

- As close as possible to those created by humans
- Accurate and thoughtful
- Meaningful to an affected user
- Well-written, well-formatted – easy to read
- Consistent (humans are fallible)
- Published already localized
- Cheaper and quicker to produce than those created by manual methods