Fanny.bmp: A sophisticated Worm for Advanced Reconnaissance and Mapping out AirGapped Systems

Techncial Report
By

William martens

william-martens@protonmail.ch

February 10, 2021.

Abstract

fanny.bmp, (better known as DementiaWheel,
or as it's codename suggests, DEWH).
Is a extremely sophisticated form of a computer worm,
which – is designed to map airgapped Computerm systems
In other words, fanny.bmp, (often shortened to fanny worm, or just fanny) can get in to systems that are not connected to the internet (so, highly secure facilities like Power Plants, like on StuXNet) and both recon the facility, trough a very complex USB-Backdoor (more on this later on the "AGENTCPD.DLL" part.) that covertly stores info about the systems it have compromised.

The 5 "major" questions around this is probably:

What? (What is it?
What vulnerabilities does it exploit?,
what kind of vulnerabilities is this?)

How? (How does it work?,
How does it spread?,
How would detection be done?,
How does it hide & store as well as communicate to C&C's if it's inside a airgapped system?)

Why? (Why make a worm like this?
What is the purpose of the worm?,
Why the level of sophistication?)

Where ? (any specific targets?)

When ? (When did it first surface?
And when did it get discovered?)

The goal of this technical report is to answer all of the above questions,
as well as give you an insight of how this would look like – with both POC's(Proof Of Concepts) and malware-samples of fanny.bmp, comparisons of StuXNet's "version" of the CVE's and Fannys "versions" of them.

As well as some links to how this (in practice) would be possible to implement
and as how to re-construct some parts of fanny).

First words from the author

Hello,
first of all I want to thank you for reading my report.

As well as to say that - this is my first “serious” technical report.
So, which is why please: any feedback, questions, or suggestions as well as improvements and/or ideas for future reports(projects) would be greatly, greatly appreciated!

If you have anything to add/ask, as above stated;
– contact me via mail.

Contents At A Glance

1 Analysis
2 (optional read) Technical Details of the LNK Exploit

4.1 Table of IOC (Indicators Of Compromise)

4.2 POCS (Proof Of Concepts)

4.3 Methods of Detection

1. The classic way
2. MetaSploit

5. Conclusion

1. Final Words

6. References

7. Bibliography

8. Index

8. Terms Index

Throughout this report I will adher to specific terms, or "words" and "codenames" as it's called. I thought of providing them in 1 page (like below) to make it easier for you as a reader.

USB-Sticks/ThumbDrives	Portable Storage.
AirGap	A system/computer(-)/network that is not connected to the internet.
APT	Advanced Persistent Threat
POC	Proof Of Concept
EQGRP	The APT EquationGroup
Fanny	The malware Fanny.bmp
OlympicGames/Stuxnet	The malware (the codename and just "StuXnet")
HPA	Host Protected Area
Drv	Driver
0day/ZeroDay	A unknown vulnerability to the vendor.

Windows File[.]Extensions

.doc,.txt	Document / Text file
.exe	executables.
.sys	System – File (Possibly a Driver)/has +SYSTEM Attribute on it.
Autorun.inf	A file often on USB-Sticks, media files automatically – – play at insertion.
.dll	
.ico	Icon files, used for e.g Applications to show icon.
.bmp,.png,.jpeg,.jpg	Media files (Pictures)
.mp3,.mp4	Media files (videos)

Introduction

1. Fanny – what is it?

Fanny is a highly sophisticated worm, with *included* modules designed to achieve several things:
A USB-Backdoor (to run even if offline, in e.g a airgapped system, most likely inside a highly-secure facility.)
Several information gathering & exfiltration – as well as command carrier, and actually a counter to stop the spreading when a critical threshold is reached to limit it's own killchain[1].
A USB-Spreading Mechanism, which copies and makes slight modifications to itself, (for config patching, and most liekely incrementing the killswitch by some integer)

- therefore, fanny was active, very long – ~ approx. 4 - 5 years it seems it have been active. By the writing of this article the exact number of infections isn't known.
But researchers estimate it is around approx. 50 000 Infections[2].

Fanny has created (as StuXNet) a new era of malware.
Namely USB-Backdoors.[3]

When was it discovered?

The forum post
Fanny was first sighted by a user on a forum, who posted about it.
According to the user on the forum[4] - "dkk" - the malware "fanny.bmp" could infect machines without autorun being enabled.

The Research around StuXNet (Olympic Games)
According to securelist, researchers discovered fanny.bmp – during some StuXNet analysis in 2010.
According to securelist, while in 2010[5] researchers that studied StuXNet noticed that fanny.bmp used the same Exploit. Not one, but two. Same exploits which was later found to be used inside StuXNet.[6]

1 Fanny-Worm-Has-Been-Freely-Available-for-Download-for-Almost-Five-Years - at softpedia
2 https://securelist.com/a-fanny-equation-i-am-your-father-stuxnet/68787/
3 USBBackdoors - at securelist
4 fanny.bmp - virus without autorun.inf - at forum.lowyat.net
5 Stuxnet - at Wikipedia
6 A Fanny Equation: "I am your father, Stuxnet"- at securelist

2 Payload of Fanny.bmp

Fanny is a worm, that – to many researchers surprise, has multiple connections to both a very sophisticated APT (Advanced Persistent Threat) Group, namely EquationGroup. Or EQGRP for short, and to other malwares such as StuXNet.

The purpose of fanny is quite clear.
-And as well as the origins is also, quite clear. Although there is no 100% evidence;
-we can still speculate.

2.2 Stealth and Persistence

KillSwitch Mechanism
To stay undetected longer Fanny utilizing a killswitch mechanism.

A counter which, is stored in a [config] – which is also the modified version of the copies fanny makes of itself when infecting a new system – when reached a critical threshold, it stops spreading.

In other words, because of this Fanny could stay under the radar for so much longer, and – fanny didn't really *destroy* anything, (that is not what it's designed for).

It's designed for advanced reconnaissance (I suppose) on AirGapped – probably highly secure facilities.

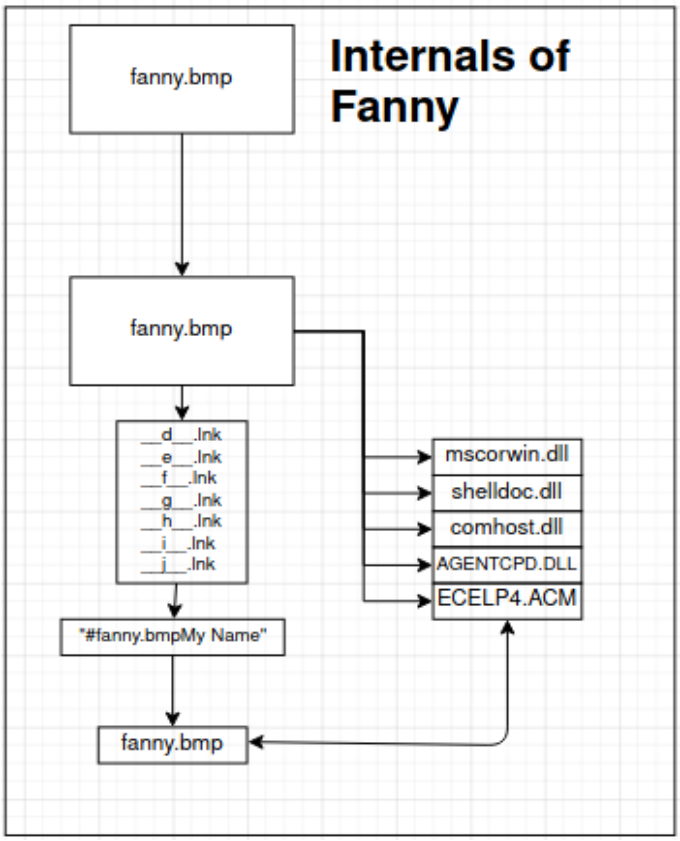
Most likely a target country/and/or a specific individual facility. Worms – tend to spread, fanny had ways around that to limit it's own spread. This is a new category in malware. A Recon tool for advanced operations(stated above) trough USB-Exfiltration. And even using the USB as a carrier for this.

3 The Internal workings of Fanny

Level of Sophistication

Fanny is highly sophisticated (especially for it's time)
As It makes use of different (yet powerful, and extremely clever) methods for both spreading, storing and exfiltrating not only information (gathered from reconnaissance)

- It also uses a state-of-the-art self-kill/countdown-kill mechanism).



Covert Comunication

As StuXNet is considered the first digital weapon,
Fanny.bmp is considered as a new kind of Malware too, namely USB-Backdoors.
Fanny is capable of covertly communicating, both receiving and sending – as well as gathering information on the compromised systems – spreading trough USB-Sticks as well as using the USB-Stick to create a USB-Backdoor on it. (It creates covert areas on the USBStick which is made from a customized/specialized FAT16/FAT32[7] FileSystem driver. [ECELP4.acm]
More on this in the **IOC** – Part.)

3.2 Relations to other malwares and APT groups

Connections to EquationGroup

EquationGroup or EQGRP often shortened to, is a highly sophisticated APT group, which – to many researchers knowledge, has connections to multiple advanced platforms, frameworks and toolkits designed for complex reconnaissance, penetration – and exfiltration of sometimes, highly secure and (by the time writing this report - likely) **airgapped**computer systems inside facilities.

7 FAT – File Allocation Table FAT - at Wikipedia

3.3 Exploits used in Fanny both of which was later found in StuXNet

Fanny used (surprisingly) 2 **zero days to both replicate and achieve extreme persistence**.

Those are:

1.) CVE-2010-2568 – A vulnerability in Windows LNK files [8]
which was later found to be used in StuXNet.

2.) MS09-025 – A Kernel vulnerability in Windows which allowed "Elevation of Privilige" [9]
– Also used in StuXNet.

The following text is the lnk files in raw(binary) form (excluding the unprintable data)

d:\fanny.bmp
My Name

e:\fanny.bmp
My Name

f:\fanny.bmp
My Name

g:\fanny.bmp
My Name

h:\fanny.bmp
My Name

i:\fanny.bmp
My Name

j:\fanny.bmp
My Name

I't should be worth mentioning that in the binary one can see the strings
?_Q_.lnk and similar patterns. Probably meaning something syntax-like:
<Driverletter>_<sameDriveLetter>_.lnk (or, in some rare cases, in Windows 2000 I noticed it wasn't using _.lnk but _.pif in some cases, but that's unusual still – in Windows 2000.)

Fanny also implements Different methods of both infecting, Payload-preparing(dll configs , paths, ..., filterX where X is a integer used in the Registry (refer to the IOC part for more information), ..., etc).

By customizing different behaviour individually for different Versions of Windows.
The LNK Files are actually *not the same / or slightly modified StuXNet LNK files, but StuXNet uses another kind of (enchanced) version of fanny.bmp's LNK files.*

Still – it's worth mentioning that one should not confuse "kinds" of lnk files, the exploit is still CVE-2010-2568 – but of course, the resulting .lnk/.pif file can differ.

In some cases, it might be a good idea to first check the system – and configure the output lnk files the resulting output of the check(s) – instead of using hard-coded lnk file(s) (which, to my knowledge – it has ***not***

8 Microsoft Windows - Shell LNK Code Execution (MS10-046) (Metasploit) - at EXPLOITDB
9 MS09-025 Vulnerabilities in Windows Kernel Could Allow Elevation of Privilege (968537) - at cvedetails

failed(although it have crashed my XP, but it was a VM so I am not sure if it was because of that.) - anyway, even if it does not crash the system, it could very likely slow things down – and I guess the authors of fanny have thought about this, because of the changing behaviour of Fanny in different versions of Windows).

4. Technical analysis

The 0day LNK/PIF Exploit

does CVE-2010-2568[10][11][12][13] sounds familiar?
If not, CVE-2010-2568 is a 0-day vulnerability in the handling of lnk and PIF files, in Windows.

If it does, of course - StuXNet used this exploit – but fanny also uses this!
And, that is just the beginning.

The Other 0day Exploit MS09-025

it seems StuXNet, fanny, and FLAME – possibly others,
Like Turla – For those who know what Turla is; for those who don't – it's a APT like EquationGroup, although – probably a bit less sophisticated. Turla is probably related to the malware AGENT.BTZ [14](that, mimics fanny and stuxnet in remarkable ways) Turla likely also knew about this bug, too. [15]
At this point I should mention again that, remember – fanny was *before* ***StuXNet***. This can only point to one thing, the authors of fanny.bmp knew about this – so did the StuXNet devs.

And – looking at the code similarities from fanny.bmp and StuXNet – a pattern of minimal modification is revealed. Which means I guess the devs of StuXNet, maybe FLAME, maybe more – devs are working closely together, or shares/have access to the same repository of vulnerabilities.

Technical Details of the LNK exploit

It's worth mentioning that this section of the paper is optional.

New readers could skip this part.

Hexdump of the [dot]LNK file used by Fanny[16]:
4c0000001140200000000000c000000000000046810000000000000000000
0000000000000000000000000000000000000000000000000000000000000000
0100000000000000000000000000000003e0414001f50e04fd020ea3a6910
a2d808002b30309d14002e002020ec21ea3a6910a2dd08002b30309d1404
0000000000000e000000653a5c66616e6e792e626d7000004d79204e616d
650000000000000000000000000000000000000000000000000000000000
Decoded:

10 Vulnerability in Windows "LNK" files? - at isc.sans.edu
11 CVE-2010-2568 - at CVE-Search
12 back-to-stuxnet-the-missing-link - at securelist
13 CVE 2010-2568 - at ExploitDB
14 Agent.btz: a Source of Inspiration? - at securelist
15 Anatomy of Turla exploits - at virusbulletin
16 All Fanny modules - at GitHub

...< a lot of raw(binary) – unprintable data > ..
...e:\fanny.bmpMy Name

Hexdump of the [dot]LNK file used by StuXNet[17]:
Hexdump too big to have here.
Decoded:

...< a lot of raw(binary) – unprintable data > ..
...\\.\STORAGE#Volume#_??
_USBSTOR#Disk&Ven_Kingston&Prod_DataTraveler_2.0&Rev_PMAP#5B6B098B97BE&0#
{53f56307-b6bf-11d0-94f2-00a0c91efb8b}
#{53f5630d-b6bf-11d0-94f2-00a0c91efb8b}
\~**WTR4141.tmp**

*(~**WTR4141.tmp is part of StuXNet's code**[18])*

4.1 Table of IOC (Indicators Of Compromise)

If one finds any of these files on their system, and/or lnk files called "_d/e/f/g/g/h/i/j_.lnk" *it's very likely that the system is compromised by a very sophisticated – worm, fanny.bmp. And that all the USB Sticks that have previously been connected to the system, shall be assumed to have fanny on them too. As well as the systems that have been transfering data between those USBs.*
Immediate action should be taken, as it's not only capable of reconnaissance – it can receive and execute, as well as be sent new commands from c&c – if not trough the internet, trough the USBSticks.

FILENAME+EXTENSION	OPTIONAL NAME COMMENTS
AGENTCPD.DLL	**USB-Backdoor** A highly sophisticated USB-Backdoor, it's purpose is to do mass-reconnaissance on airgapped systems. In other words, the AGENTCPD.DLL is the part of fanny that creates, and handles some of the USB-Spreading, but also for the allocation on ew USB sticks. It achieves this by implementing it's very own FAT16/FAT32FS Driver.

17 Stuxnet code - at Archive
18 SilverPoision stuxnet-source-code - at GitHub

	Inside the binary, one can see it performs some operations on the following (and more) Registry Keys: \System32System\CurrentControlSet\Services\USBSTOR\Enum System\CurrentControlSet\Services\PartMgr\Enum Trough some string analysis, one can see the .bmp extension of the rootkit(**not agentcpd.dll**) \\.\a: FILENAME *.bmp \restore\ DLL: It's Export (EntryPoint) is: _Start@16 0
comhost.dll	**USB SPREADING** Includes configurations and other information of the worm's behaviour. Includes the kill-switch counter. When counter reaches a critical threshold, the [current] worm stops spreading – limiting the detection of the worm. Sometimes communicates between mscorwin.dll and is responsible for the USB-Spreading. DLL: It's Export Ordinal 4 is: dll_installer_4
ECELP4.ACM	**FAT16/32 DRIVER – USB Allocator & Persistence** Tool Is a FAT16/32 Driver and a handler for processes, their names to the path of the executables. Registry Artifacts can be found at (But is not limited to):

	*HKLM\System\CurrentControlSet\Control\MediaResources**acm****ECELP4****filter2*** *HKLM\System\CurrentControlSet\Control\MediaResources**acm****ECELP4****filter3*** *HKLM\System\CurrentControlSet\Control\MediaResources**acm****ECELP4****filter8*** ***and possibly more (like filter9, ...) should be checked as well.*** Trough some string analysis, one can see the name ECELP4.ACM as well as it is a driver, **DriverProc** and **filter** is easily notable. ECELP4.ACM DriverProc filter System\CurrentControlSet\Control\MediaResources\acm\ecelp4
fanny.bmp	The "main" module. This is the actual malware that was discovered. Along with the lnk files. It set's file attributes SYSTEM | HIDDEN on the lnk files, and several of the other files included in fanny.bmp. Uses MS09-025
mscorwin.dll	**Utiller/Tool** For the handling of critical processes like LSASS (Local Security Authority Subsystem Service) Is sometimes communicating with comhost.dll.
shelldoc.dll	**Rootkit -** shelldoc.dll is the rootkit-part of fanny. It hides files that it detects is related with fanny trough removing them from the SysListView32 in Windows Explorer[19] (.bmp files, fanny-name-related files, (and folders), .lnk files as well)
Lnk Files	Used to load fanny.bmp trough the LNK exploit (CVE-2010-2568)[20][21]

4.2 POCS

If you want to experiment and analyze yourself,
samples of StuXNet is available online, and I have – by the time writing this report,

19 ListView Control Overview (Windows Forms) - at docs.microsoft.com
20 stuxnet-incident - at zscaler
21 cve 2010-2568 - at exploitdb

published fanny.bmp (as well as it's LNK files, and the modules – AGENTCPD.DLL, shelldoc.dll, comhost.dll, fanny.bmp, ECELP4.acm, msupdate.exe, and the TMP file ~DE1923.tmp) on GitHub.[22]

POC Video 1: Fanny.bmp[23]

POC Video 2: Agent.btz[24]

4.3 Methods of Detection

How to detect or know if a system is infected with fanny?
There are several ways, first the obvious ones:

Method 1: the classic way

boot into a trusted-live-OS (Linux is a good choice) from a LIVE-USB, offline – (Trusted – as the USB has never been inserted in a system which is suspected to have been compromised by either **fanny but not compromised in general**). And check the files there, mount the hard disk (the compromised system's hard disk(S) if there are multiple) and check the **IOC section.**

Method 2: MetaSploit.

MetaSploit can't only reconnaissance and penetrate systems, but did you know you can check for e.g malware with MetaSploit too? For example you can detect DUQU Artifcats using a module[25] in the MetaSploit-Framework[26].

I have created a module that has the goal of detecting[27] the Registry Artifacts made by fanny -
In the registry.[28][29]

5 Conclusion:

During my analysis, I had in mind to create something to detect (and, later – re create certain parts of) fanny and it's related files.
I noticed how fanny Creates several registry keys on the compromised system. And decided to write a Module[30] (in the Ruby[31] programming language) using MetaSploit to detect this.
Fanny - is still (by the time I writing this) still quite unknown to many.

I hope by the time you read this report, that fanny.bmp is widely known, publicly.
Thanks for reading this report, and any feedback is very well welcomed!

22 fanny.bmp source - at GitHub, November 30, 2020
23 POC video 1 - at Youtube
24 POC video 2 - at Youtube
25 Windows Gather Forensics Duqu Registry Check - at Rapid7
26 source code of duqu_check.rb - at Rapid7's GitHub
27 CVE-2010-2568 - at Attacker KB, Last updated July 30, 2020
28 FannyBMP or DementiaWheel Detection Registry Check - at Rapid7
29 source code of fanny_bmp_check.rb - at Rapid7's GitHub
30 MetaSploit Fanny.bmp Detection Module Source code - at GitHub
31 ruby - at ruby-lang.org

Final Words:

What did you like the most?
What did you dislike?
Something to add, or change in the report?
A question or feedback?

Send them all in to william-martens@protonmail.ch
Discord: Ken-Kaneki#3978

6. References

Sites and Articles:

Springer Link (for research papers)
Securelist
Wikipedia
Zscaler
ExploitDB
Youtube
Antivirus Providers like Kaspersky
attackerkb (as an additional cve-2010-2568 source)

7. Bibliography

Books and Text's:

Eder-Neuhauser, P., Zseby, T. & Fabini, J. Malware propagation in smart grid networks: metrics, simulation and comparison of three malware types. J Comput Virol Hack Tech 15, 109–125 (2019). https://doi.org/10.1007/s11416-018-0325-y

Practical Reverse Engineering by Bruce dang – Introduced me into all of this with reverse engineering.

K&R 2nd edition by Brian Kernighan and Dennis Ritchie – really got me into this with C programming.