SPOOKS

COLLABORATORS

	TITLE : SPOOKS	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 31, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

SPOOKS

1.1

```
I decided to make file with text about new stuff because
we are recently flooded with such shit.

   Spooky stuff: 10.4.2001

  4ef9 trojans on Aminet!

  New linkvirus! (BASTARD)


   Spooky stuff: 27.4.2001

   BASTARD INSTALLER FOUND!

   NEW LINKVIRUS AND IT'S INSTALLER (BOBEK)

   ABOUT FILE BASTARD VIRUS IS LOOKING FOR
```

1.2 bast

```
There appeared brand new link-virus. It is well coded, and seems
to have nothing to do with the work of lamer described above.
Here is the first analyze of that virus.
At the moment the range of spreading is unknown,
but I heard the installer is an archive with pointers or somthing
in this kind. Jan Andersen of VHT-DK is working on it or already finished.

             BASTARD LINKVIRUS ANALYZE


The virus doesn't seem to be able to spread on so many machines,
```

but of course file removals will be ready as soon as possible.

1.3 4ef9

This text says some words about trojans recently made by UAE lamer
who thinks that having 4ef9 linker makes him hacker.
Such lame trojans have been seen about five years ago,
but now addtional code instead of mannipulating BBS:user.data
tries to insult innocent via internet.
This is the only difference during the years, this means the lamers
still on the same level.

```
BlazeWCP.lha    32862 bytes the file 'BlazeWCP' had been
        linked with shitty e-mail sender,
                                and the file version got faked
        said to be: v1.8

FBlit.lha  142086 bytes the file 'FBlit' had been linked
        with shitty e-mail sender,
        and the version got faked
        said to be: v3.84

StackAttack.lha   69229 bytes the file 'StackAttack' had been linked
        with shitty e-mail sender,
        and the version got faked
        said to be: v1.2b

Safe.lha    20737 bytes the file 'Safe' had been decoded,
                                version had been changed, then code was
        linked with the same shitty e-mail sender,
        and crunched until size reached 7000 bytes.
        Lamer used quite old Safe. Maybe he was
        thinking about attack from longer time.
                                This archive was replaced with correct Safe,
        as soon as it was possible (Thanks Error!)
        said to be: v14.10
```

Please check Your system with e.g. VirusZ and note that,
4eb9 or 4ef9 linker can't mean anything good to you.
Those linkers were used to make trainers or add crack intros,
but also (surprise!) were often used to make trojans.
Experience learns that linked code always does more harm than good.
If You were watching Your fresh stuff with good (even old)
AV software You would easily see such strange things.

The linked code is 940 bytes long and is EOR crypted.
Same code was used to make all four trojans.
Safe v15.2 is able to stop processes created by this added code,
so email won't be sent every 60 seconds.
Linked part tries to send some insulting text.
After decoding You can see inside such shit:

```
(...)
0290  aage-partner.com
02A0  >..DATA..From: M
02B0  OS Rul3z y0u bit
02C0  ch! <>..Subject:
02D0   MorphOS - The R
02E0  eal Slim OS4ady.
02F0  ..Fuck U JERKIN
(...)
```

The e-mail is sent to haage&partner and contains some insulting text.

Note to lamer:
What da fuck the 'memory leak is'?
I don't have UAE, so I am wondering about that...

Thanks to Jan for sending all the archives, and to Error for
keeping eyes opened.

1.4 point

Pointers.lha 6874 bytes long...

...is the installer for the BASTARD LINKVIRUS!
The executable is hidden inside installer script and I must
admit I haven't seen such thing before.
It was done (in very clever way) with special tool which
changes binary to valid installer script data.
This can be seen as real MACRO virus for Amiga!

NOTE:
There was no script icon, so I think almost noone installed
the virus!

This installer script generates file called RAM:temp,
which is stonecracked executable with BASTARD virus.
This is just THE FIRST file of virus. It contains also
some text and even the name of the virus:

Antidisassemblishmentaryonism v1
(I think everyone still use the name I have invented :-)

There was nothing new in file beside that additional text.
It also says about the authors, which are not the same people behind
those lame 4ef9 trojans (I came to this conclusion only
by watching the code, so You see the differences was large.).
As always I will not publish the text inside not to satisfy
virusmakers even this is done very clever and not to
infect so many machines.

Thanks to Jan Andersen for finding all recent archives
with pointers!

1.5 bob

```
The new linkvirus appeared!

It is called BOBEK and seems to be
work of the authors of BASTARD. It is done as simple as possible,
but the some parts of the engine are comparable or equal
to BASTARD and even Motaba-3.
I don't know what authors had in mind. Maybe to make STORM to release
something else?

BOBEK LINKVIRUS ANALYZE

Installer of BOBEK is OzzeAga.lha which is already kicked out from
Aminet. I got the archive recently and saw that infected file
is the BoardMaker. What is more important:
the main game file seems to be CRACKED to use littlebit faked
keyfile (the visible text about Amiga Community).
I think the keyfile system from original game should detect such changes.
I found SEEM TO HACKED code after some comparing instrutions,
but keyfile system is quite complicated, so I can't tell if really.

To the authors of Ooze:
Is it possible to check faked keyfile? Maybe there still hidden
infomation about who bought this copy, because the body of keyfile
seems to be original part of correct key!

In this place thanks to guy who discovered the BOBEK linkvirus:
Ingo Foerster.

And to guy who dicovered it's installer:
Frank Niewiedzial



The only solution to clean the files is newest Virus Executor!
```

1.6 bastard

```
Entry..............: BASTARD (temporary name)
Alias(es)..........: -
Virus Strain.......: Motaba(?)
Virus detected when.: 4.2001
              where.: internet
Classification.....: Linkvirus,memory-resident, not reset-resident
Length of Virus.....: 1. Length on storage medium:     c.a.2100 Bytes
                      (uses polimorphic engine)
                      2. Length in RAM:                    8192 Bytes

-------------------- Preconditions -----------------------------------

Operating System(s).: AMIGA-DOS Version/Release..: 2.04 and above (V37+)
Computer model(s)...: all models/processors (MC68000-MC68060)
```

```
---------------------- Attributes ---------------------------------------

Easy Identification.: none

Type of infection...: Self-identification method in files:

                      - checks first byte of first codehunk for $61
                        (part of jump to viruscode)

                      Self-identification method in memory:
                      - indirectcly the virus is aware of itself:
                        * checks for $-1 in tc_userdata
                          field of every process, this value
                          is stored by exec/TaskWait list scanner,
                          already checked processes are skipped
                        * the try to hack asl.library fails,
                          so memory is freed

                      System infection:

                      -  tries to guess paths to runned programs
                         via pr_Homedir and task name.
                         This gives about 2-5 valid filepaths
                         (mainly in WBStartup) to infect.

                      -  Tries to hack in memory code of AllocRequest
                         of asl.library with patch that tries to
                         hack VirusCheckerII process (gets
                         via seglist Open call of this killer and
                         patches it!). I don't know
                         which version(s) author of virus had tested.

                      Infection preconditions:

                      - File is between 2000 and 32000 bytes
                      - Hunk Code is found
                      - File is not infected already
                      - device is validated
                      - device contains free blocks

Infection Trigger...: 1. Accessing files via checking them with VirusCheckerII.
                      2. Direct infection of some runned programs
                         after run of an infected file.
                      Files containing a "l" or "L" or "-" or "V" or "v"
                      will be not infected.

Storage media affected:
                      all DOS-devices

Interrupts hooked...: None

Damage..............: Permanent damage:
                      - Crashes system.
                      Transient damage:
                      - none
Damage Trigger......: Permanent damage:
                      - File ENV:mui/spirit.1.prefs exists
```

```
                    Transient damage:
                    - none

Particularities.....: Polimorphic decrypt routine.
                    The decryptor is 256 bytes long and before
                    it is always: movem.l d0-a6,-(sp)
                    This engine is (for me) a new one, but doesn't
                    contain enough stuff to prevent "checksum"
                    detecting of the infected files.
                    The truth is even better. We can decode virus
                    using the technic found inside it
                    (the crypter and decrypter are same!).
                    The polimorphic engine always contains
                    one loop, one eor, one move.l 4.w,a6,
                    two lea.l rest are random moveq and shitfs
                    like lsl.l #2,d4 etc.
                    The decrypt algo may vary
                    if in the decrypt loop appear random
                    instruction that changes cryptkey register,
                    I didn't get any crashing example.

                    The virus replaces first longword of the
                    first codehunk with bsr.w to virus code.
                    The original value is restored by
                    decrypted virus code. And the stack will
                    be mainipulated to call the program first
                    and then call the main virus code.
                    Note that there is no detailed check for
                    this long, so every file without $61
                    at the begin will be infected.
                    This means also that files with reloc instruction
                    in first long will cause guru after infection.

                    New ideas at all. The virus looks excellent
                    compared to Motaba-3 that is supposed to
                    be the base of this viral engine.
                    Direct hacking of things that are ram only
                    is problematic subject and there is incredibly
                    large amount of things that can be hacked
                    in future in the same way.

                    One of these bastards that
                    if run from an icon will not crash
                    with the wellknown GURU 87000004. Thats because
                    of the executing of virus code AFTER program.

Similarities.......: Link-method is first hunk increasing.
                    The main code is comparable to motaba-3.
                    Length polymorph is same!
                    The change of lenght is depending on
                    'a' in filepath.
                    The path creator is idea comparable to
                    Antonio and PolishPower viruses.

Stealth............: FindTask must be pointing to $fxxxx or virus
                    will not try to hack VCII.
                    Open must be pointing to $fxxxx or virus
```

```
                      will not perfom any action.
                      Write must be pointing to $fxxxx or virus
                      will not perfom any action.
                      Lock must be pointing to $fxxxx or virus
                      will not perform check for ENV:mui/spirit.1.prefs.
                      The virus doesn't patch ROM library vectors,
                      and the hackings of VC and asl.library are done
                      in quite tricky way.

Armouring...........: Polymorphic decryptor is used, length
                      of added code is changing in small range and
                      at the end of the virus is more or less garabage.
                      The virus contains some of the popular tricks
                      like bsr and then increasing sp to mix code
                      with data and some confusing/antidisassembling
                      instructions.

Comments............: -

--------------------- Agents ----------------------------------------

Countermeasures.....: -
above Standard means......: -

--------------------- Acknowledgement -------------------------------

Location............: Pawlowice, Poland  4.2001
Classification by...: Zbigniew Trzcionkowski
Documentation by....: Zbigniew Trzcionkowski
Date................: 4.2001
Information Source..: Virus disassembly and reverse engineering
Copyright...........: This documentation is public domain

====================== End of BASTARD ====================================
```

1.7 bobek

```
Entry...............: Bobek!
Alias(es)...........: -
Virus Strain........: -
Virus detected when.: -
              where.: internet
Classification......: Linkvirus, memory-resident, not reset-resident
Length of Virus.....: 1. Length on storage medium:          460 Bytes
                      2. Length in RAM:                   65535 Bytes

--------------------- Preconditions ---------------------------------

Operating System(s).: AMIGA-DOS Version/Release..: 2.04 and above (V37+)
Computer model(s)...: all models

--------------------- Attributes ------------------------------------

Easy Identification.: visible text '[BOBEK!]' in every infected file
```

```
Type of infection...: Self-identification method in files:

                      - compares length declared in hunkheader
                        with the real length (this also
                        avoids infection of some crunched files)

                      Self-identification method in memory:
                      - checks for $0 in libOpen address of exec.library.
                        Exec base is available by second
                        longword of memory, so this routine
                        isn't used in normal system. This address is LOST
                        forever (till reboot)!
                        Note that this makes VT-Schutz not work.

                      System infection:
                      -  infects ExNext function of dos.library
                         The paths to infect are made with
                         NameFromLock and filename in returned FileInfoBlock.
                         This gives in some cases wrong paths,
                         so some directories won't be touched by virus.
                         The patch is done in way that prevents VirusZ
                         from invitating user to VectorCheck.

                      Infection preconditions:

                      - File is between 1000 and 32000 bytes
                      - Hunk Code is found
                      - File is not infected already
                      - device is validated

Infection Trigger...: Scanning directories (filemanagers,
                      filerequesters etc.).
                      Not all files that could be infected will
                      be infected at once.

Storage media affected:
                      all DOS-devices

Interrupts hooked...: None

Damage..............: Permanent damage:
                      - none
                      Transient damage:
                      - none
Damage Trigger......: Permanent damage:
                      - none
                      Transient damage:
                      - none

Particularities.....: The virus code is highly optimized and looks
                      like work of very experienced assembler programmer.
                      The linker is made with one Open/Close,
                      so there is no so much noise like with some
                      other viruses.
                      The code isn't crypted and the linked part
                      has always static size 460 bytes.
                      Memory allocation is done only once at start because
```

of checking small range of filesizes.
The infected file has always replaced first longword
of first code hunk with BSR.W to init code which
runs the program and then the virus itself.
Due to above there is no problem of GURU
after running from Workbench.

The programs with reloc entry at the replaced
long may crash, but I couldn't find such example,
because:
there is test for $4E at the first LONG,
so files with such jumps will be untouched
(including the reloc ones: 4EF9 and 4EB9).

Similarities.....: Link-method is first hunk increasing.
Used known code to check if the Open vector
points to ROM.

Stealth.....: The cheating of VirusZ's VectorCheck is done
with patchformat known from MCP and PatchControl.

The virus puts the new infected length
to FIB. In the other words the ExNext
always returns the real size of file.

The known from Motaba-3 trick is used
to check if Open points to ROM.
Due to that no virus action is visible
in SnoopDos or DosTrace.

There are used two additional technics to
decrease the noise while scanning directories:
- check if size is dividible by 4 (executables)
- infect try will be performed depending on
  state of bit 0 of $dff007
IMHO it isn't good enough.

Armouring.....: Nothing special.

Comments.....: The virus contains VISIBLE string:

'[BOBEK!]'

NOTE!
1. There is no code to restore filedate
   after infection.
2. Virus is not crypted.
Maybe someone forgot to release it few
years ago... :-)

The best way to switch the virus off in file
is to change visible string 'dos.library' few
bytes before [BOBEK!] to anything else.
Old Filemaster v2.2 is enough!
Xvs.library will be aware of that.

---------- Acknowledgement ----------

```
Location.............: Pawlowice, Poland  4.2001
Classification by...: Zbigniew Trzcionkowski
Documentation by....: Zbigniew Trzcionkowski
Date................: 4.2001
Information Source..: Virus disassembly
Copyright...........: This documentation is public domain

==================== End of [BOBEK!] ================================
```

1.8 spir

```
As some of You have seen in Bastard analyze the virus is looking
for file called:

ENV:mui/spirit.1.prefs

This is of course prefs file generated by wellknown (?)
tool to make keyfiles/crack software made by DigitalCorruption.
So the virus judges if You support piracy and crashes
the system is so :-)


Thanks to Henner Puderbach for information.
```