

# Nigilant32 for First Responders

## Active Memory Imaging

Using the Image Physical Memory tool within Nigilant32 a first responder can capture the contents of volatile active memory (RAM) and save it to a network share or attached portable storage device for future analysis.

Nigilant32 is an incident response tool designed to capture as much information as possible from a running system with the smallest potential impact. Nigilant32 has been developed with Windows 2000, XP, and 2003 in mind, and should work fine with computers running one of those operating systems.

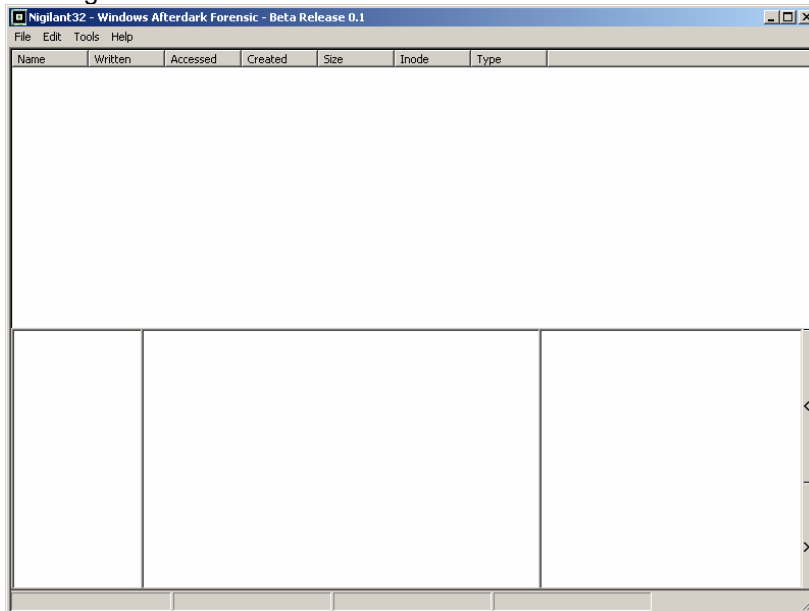
Let's look at an example of imaging physical memory.



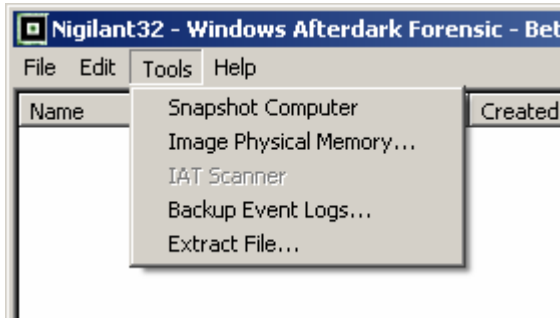
*This article is the last in a series of publications developed to assist the first responder in understanding the many uses of Nigilant32 <n-eye-jill-ant32> in an incident response capacity. In this article we cover using Nigilant32's active memory (RAM) imaging engine to capture the contents of volatile memory.*

*Nigilant32 was developed by Agile Risk Management LLC, the most recent and update to date version can always be found at [www.agilerm.net](http://www.agilerm.net).*

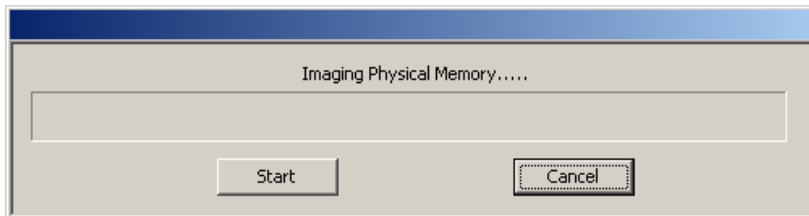
### The Nigilant32 main console



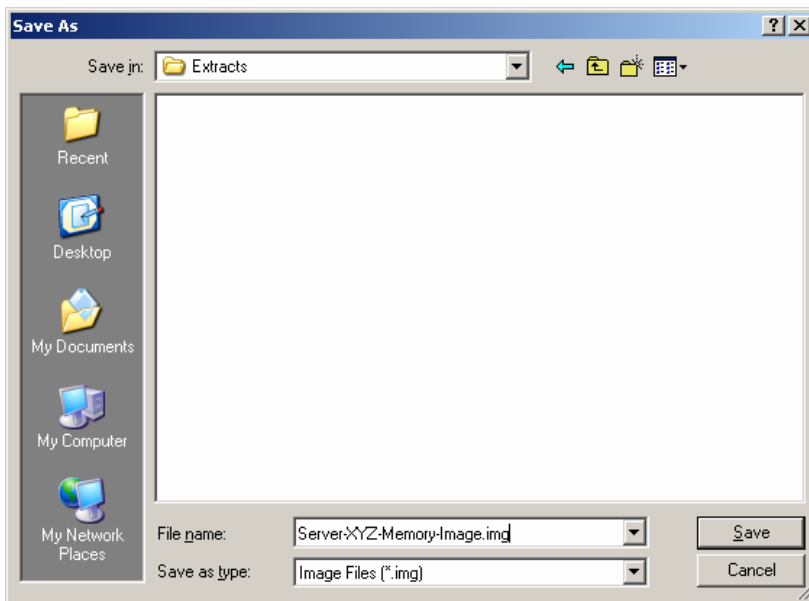
In order to image physical memory we must use "Image Physical Memory..." from the Tools menu..



Let's review the output of the Image Physical Memory... command.



By selecting "Start" we tell Nigilant32 to open a File/Save dialog to determine where we would like our memory image placed. It is important to select a destination that has sufficient space to hold the memory image generated. The image will be identical in size to the total ram on the workstation/server. Physical memory imaging has only been tested on systems with 2 Gigabytes of active memory or less, we do not know how it will perform on larger systems.

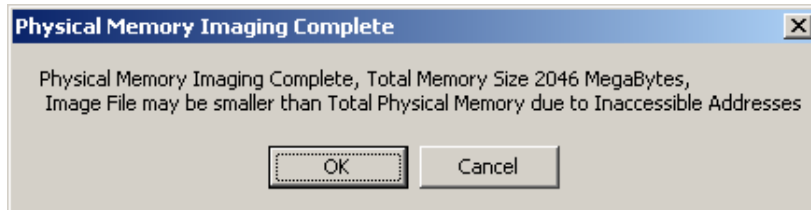


You will need to assign a name to your memory image file, in our example we choose "Server-XYZ-Memory-Image.img". Select "Save" to begin the imaging process.

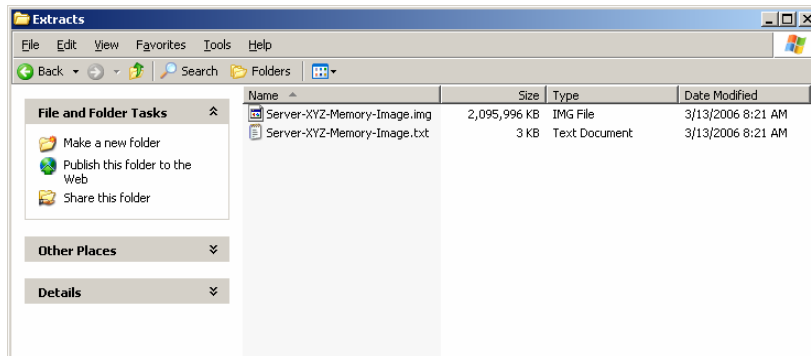
*The Sleuthkit Project is maintained and actively developed by Brian Carrier. For more information on Sleuthkit visit [www.sleuthkit.org](http://www.sleuthkit.org).*

*Nigilant32's filesystem navigation console provides information about each item, including Name, Last Write Date/Time, Last Accessed Date/Time, Last Create Date/Time, Size, Inode, and Type (Folder/File).*

As the memory image is being obtained, the progress bar will indicate the memory imaged vs total amount of memory to be imaged.



When the imaging process is complete you will be presented with the above imaging completion message. It provides the total memory size, and indicates there were portions of physical memory that could not be imaged. This is common on all platforms (Windows 2000, XP, 2003) and does not result in an un-usable memory image.



A review of the destination folder shows both the memory image (.img) as well as a text file (.txt).

The text file contains a report of the memory imaging process and includes any portions of physical memory which could not be extracted.

```

Server-XYZ-Memory-Image.txt - Notepad
File Edit Format View Help

Physical Memory Imaging Log, Total Memory Size 2145816576 Bytes
Invalid Page at Offset 0x6000, Filling output buffer with zero.
Invalid Page at Offset 0x45000, Filling output buffer with zero.
Invalid Page at Offset 0x46000, Filling output buffer with zero.
Invalid Page at Offset 0x47000, Filling output buffer with zero.
Invalid Page at Offset 0x48000, Filling output buffer with zero.
Invalid Page at Offset 0x49000, Filling output buffer with zero.
Invalid Page at Offset 0x4a000, Filling output buffer with zero.
Invalid Page at Offset 0x4b000, Filling output buffer with zero.
Invalid Page at Offset 0x4c000, Filling output buffer with zero.
Invalid Page at Offset 0x4d000, Filling output buffer with zero.
Invalid Page at Offset 0x4e000, Filling output buffer with zero.
Invalid Page at Offset 0x4f000, Filling output buffer with zero.
Invalid Page at Offset 0x50000, Filling output buffer with zero.
Invalid Page at Offset 0x51000, Filling output buffer with zero.
Invalid Page at Offset 0x52000, Filling output buffer with zero.
Invalid Page at Offset 0x53000, Filling output buffer with zero.
Invalid Page at Offset 0x54000, Filling output buffer with zero.
Invalid Page at Offset 0x55000, Filling output buffer with zero.
Invalid Page at Offset 0x56000, Filling output buffer with zero.
Invalid Page at Offset 0x57000, Filling output buffer with zero.
Invalid Page at Offset 0x58000, Filling output buffer with zero.
Invalid Page at Offset 0x59000, Filling output buffer with zero.
Invalid Page at Offset 0x5a000, Filling output buffer with zero.
Invalid Page at Offset 0x5b000, Filling output buffer with zero.
Invalid Page at Offset 0x5c000, Filling output buffer with zero.
Invalid Page at Offset 0x5d000, Filling output buffer with zero.
Invalid Page at Offset 0x5e000, Filling output buffer with zero.
Invalid Page at Offset 0x5f000, Filling output buffer with zero.
Invalid Page at Offset 0x60000, Filling output buffer with zero.
Invalid Page at Offset 0x61000, Filling output buffer with zero.
Invalid Page at Offset 0x62000, Filling output buffer with zero.
Invalid Page at Offset 0x63000, Filling output buffer with zero.
Invalid Page at Offset 0x64000, Filling output buffer with zero.
Invalid Page at Offset 0xa7d1000, Filling output buffer with zero.
Invalid Page at Offset 0xa7e1000, Filling output buffer with zero.
Invalid Page at Offset 0xa810000, Filling output buffer with zero.
Invalid Page at Offset 0xa820000, Filling output buffer with zero.
Invalid Page at Offset 0x1ccb0000, Filling output buffer with zero.
Invalid Page at Offset 0x1d141000, Filling output buffer with zero.
Invalid Page at Offset 0x1d1a1000, Filling output buffer with zero.
Invalid Page at Offset 0x1d220000, Filling output buffer with zero.
Invalid Page at Offset 0x1d280000, Filling output buffer with zero.

Physical Memory Imaging Complete.

```

The final step to memory imaging is analysis. However, there are currently no good tools or applications available to perform memory image analysis, well at least not public ones, yet.

```

Revenant - NOT FOR DISTRIBUTION Copyright Agile Risk Management LLC 2006
File Help

Processes
System
SMSS.EXE
CSRSS.EXE
WINLOGON.EXE
SERVICES.EXE
LSASS.EXE
svchost.exe
spoolsv.exe
svchost.exe
regsvcs.exe
VMwareService.exe
WinMgmt.exe
svchost.exe
explorer.exe
VMwareTray.exe
VMwareUser.exe
Nglant32.exe

Process Information
Image Name (Executable): mstask.exe
Process Identifier (PID): 516
Process Environment Block (PEB): 76df000
Directory Base (DirBase): 5b70000
Blink (BLINK): 816d10c0
DLL Path (DLLPATH): C:\WINNT\system32\C:\WINNT\system32\C:\WINNT\C\WINNT\
system32\C\WINNT\C\WINNT\system32\wbem
Image Path Name (IMAGEPATH): C:\WINNT\system32\MSTask.exe
Command Line (CMDLINE): C:\WINNT\system32\MSTask.exe
Window Title (wNDTITLE): C:\WINNT\system32\MSTask.exe
Desktop Information (DESKTOPINFO): WinSta0\Default
Initialization Order Module List

Full Dll Name: C:\WINNT\system32\ntdll.dll
Entry Point: 0
Image Size: 507904

Full Dll Name: C:\WINNT\system32\KERNEL32.dll
Entry Point: 7c57740
Image Size: 733184

Full Dll Name: C:\WINNT\system32\MSVCRT.dll
Entry Point: 78001000
Image Size: 282624

Full Dll Name: C:\WINNT\system32\RPCRT4.dll
Entry Point: 77860316
Image Size: 491520

Full Dll Name: C:\WINNT\system32\ADVAPI32.dll
Entry Point: 7c24ba5
Image Size: 413696

Full Dll Name: C:\WINNT\system32\USER32.dll
Entry Point: 77e4149
Image Size: 766656

```

The above screen capture displays Agile Risk Management LLC's internal application for analyzing physical memory images.

More information will be made available as the tool matures and evolves.

*Matthew Shannon has over seven years of professional experience in private industry, including KPMG LLP, ExxonMobil, and United Technologies. Mr. Shannon has successfully led multiple information security assessment engagements, including successful penetration of multi-million dollar financial institutions, both international and domestic. In addition, Mr. Shannon has been the lead investigator on numerous computer forensics engagements, including intellectual property theft and employment law. Mr. Shannon is also a well received speaker and author. He has instructed the United States Secret Service on specific digital forensics techniques and was a well received speaker at the DEFCON 11 annual Information Security conference in Las Vegas Nevada. Additionally, Mr. Shannon has been published in the International Journal of Digital Evidence for his work on incorporating statistical inference into digital forensics investigations as well as multiple bar journal articles and online digital forensics publications.*

*Mr. Shannon graduated cum laude from The University of Florida in Decision and Information Sciences (BSBA) in 1999. He is a member in good standing of ISSA, in addition, Matthew holds numerous professional information technology certifications, and is the developer of Nigilant32, Agile Risk Management LLC's Incident Response Tool.*

**Matthew M. Shannon** CIFI,  
CISSP  
Principal  
Agile Risk Management LLC  
mshannon@agilerm.net