

Volatility3 Plugin Development Test Report

(findpid.py, mmname.py)

Team_Digital Forensic Study

(neck392, Sooboon)

Volatility3 플러그인 개발(findpid.py, mmname.py)

2025. 05. 06.

목차

I

테스트 개요

- ① 프로젝트 명
- ② 버전
- ③ 테스트 목적
- ④ 테스트 환경
- ⑤ 테스트 셋

II

상세 테스트 내용

- ① 분석 시험

III

테스트

- ① 분석 시험 결과
- ② 결함
- ③ 특이사항

I

테스트 개요

◦ Volatility3의 사용 편의성을 개선하여 제작한 2개 플러그인의 동작 및 동작 결과를 확인하고 효율성, 정확성 및 신뢰성을 평가한다.

① 프로젝트 명

◦ Volatility3-Plugin-Development

② 버전

◦ Version 0.0.1

③ 테스트 목적

- findpid.py 분석 기능 정확성, 안정성 및 신뢰성 평가
- mmname.py 분석 기능 정확성, 안정성 및 신뢰성 평가

④ 테스트 환경

	PC1	PC2
OS	Windows 11 Home	Windows 11 Pro
프로세서	Intel i7 9 th Gen	Intel i5 13 th Gen
IDE	VS Code	VS Code

⑤ 테스트 셋

Damian.mem (.mem file)	TestSet1	TestSet2
Plugin	findpid.py	mmname.py

II

상세 테스트 내용

○ 테스트 진행은 다음과 같은 방법으로 진행되며, 분석 시험으로 진행된다.

- 1) 테스트 PC에서 Volatility3 및 해당 플러그인 실행
- 2) 실행된 Volatility3에서 findpid.py 플러그인 명령어 동작 확인
- 3) findpid.py 명령어 옵션 동작 확인 및 평가
- 4) 실행된 Volatility3에서 mmname.py 플러그인 명령어 동작 확인
- 5) mmname.py 명령어 옵션 동작 확인 및 평가

① 분석 시험

테스트 번호	함수 명	테스트 항목	상세 설명	담당자
111	통합 테스트	Volatility3 기본 동작 확인 및 사용자 정의 플러그인 적용 여부 확인	정상적인 Volatility3 실행 여부를 확인하고 추가한 2개의 플러그인을 인식하는 지 검증	김민서, 최원혁
121	def get_requirements(cls):	필수 인자 정의 및 옵션 설정 여부 확인	kernel, name, exact 등 필수 요구사항이 정확히 정의되어 있으며, 명령어 도움말에 출력되는지 확인	김민서
122	def _generator(self) -> Iterator:	프로세스 필터링 조건 및 시간 정보 추출 확인	'--name' 및 '--exact' 조건에 따라 대상 프로세스만 필터링하고, 프로세스 생성 및 종료 시간이 정확히 출력되는지 확인함	김민서
123	def run(self):	TreeGrid 출력 구조 확인 및 결과 포맷의 일관성 확인	'_generator()'로부터 받은 데이터를 'TreeGrid'로 변환하여 컬럼 및 결과 형식이 일관되게 출력되는지 확인함	김민서
131	def get_requirements(cls) -> List[interfaces.configuration.RequirementInterface]:	사용자에게 요구되는 필수 및 선택 인자 정의 정확성 확인	사용자에게 요구되는 인자들의 정의가 정확한지 확인한다. (--name, --dump, --list, --select_pid) 명령어 도움말 출력 여부 확인.	최원혁
132	def _generator(self, procs):	메모리 매핑 정보 출력 및 덤프파일 생성 테스트 (->TreeGrid 형태로 생성)	메모리 매핑 정보 출력(TreeGrid 형태) 여부 (+ 내용 정확성 확인), contextlib.ExitStack() 사용 with 문 안에서 파일 핸들링 -> pid.XXXX.dmp 형식 준수 여부	최원혁
133	def run(self):	프로세스 필터링, 옵션 조합 테스트, 예외 메시지 출력 및 정확성, 일관성 확인(실행 로직, 제어흐름 분기 확인)	실행 로직, 제어흐름 분기 확인, 예외처리 확인 (예외 메시지 출력, 디코딩 실패, Timestamp 파싱 실패 등의 예외는 개별 try-except로 처리) 작동 방식 확인(프로세스 필터링, _generator 호출 여부, PID 선택 처리, 리스트 출력)	최원혁

테스트

① 분석 시험 결과

테스트 번호	테스트 종류	테스트 항목	담당자
111	통합 테스트	Volatility3 동작 및 플러그인 적용	김민서, 최원혁
함수 명			
Volatility3 적용 여부 통합 테스트			
테스트 목적			
Volatility3 동작 및 개발한 2개의 플러그인 인식 여부 확인			
테스트 환경			
OS: Window11			
Memory file: Damian.mem			
Plugin: findpid.py, mmname.py			
테스트 케이스			
1. Volatility3 저장 경로 이동 2. “Python3 vol.py -h” 입력 3. 플러그인 인식 및 적용 여부 확인			
테스트 결과			

[illegible]

[그림 1-1] -h 옵션 수행 결과

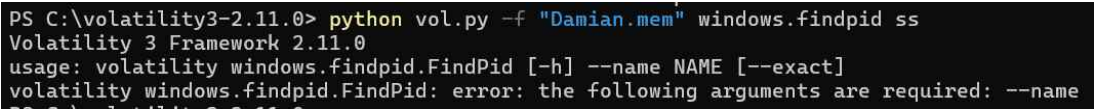
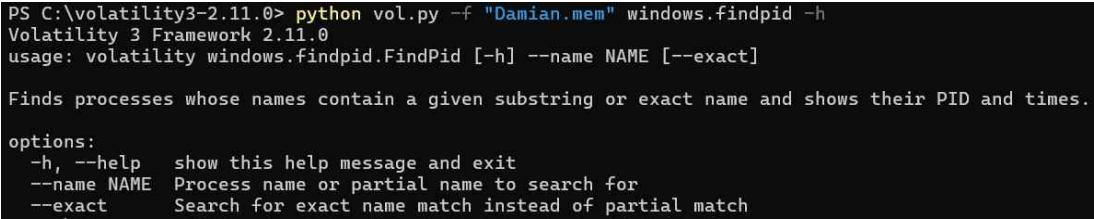
볼라틸리티에서 해당 '--help' 명령어 수행 결과 [그림 1]과 같이 개발한 2개의 플러그인 모두 인식되며 “plugins:“ 안에서도 확인할 수 있다.

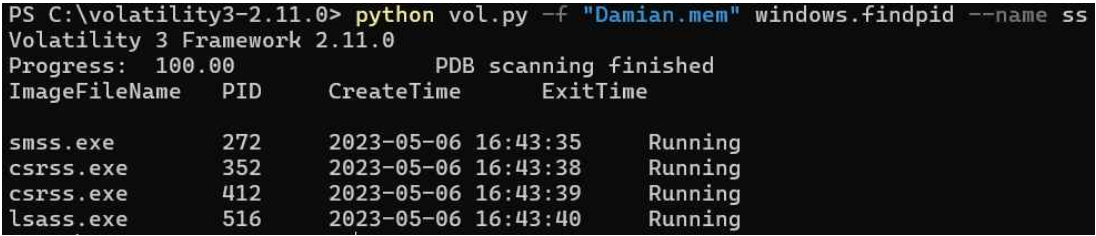

```
windows.findpid.FindPid
    Finds processes whose names contain a given substring or exact name and shows their PID and
```

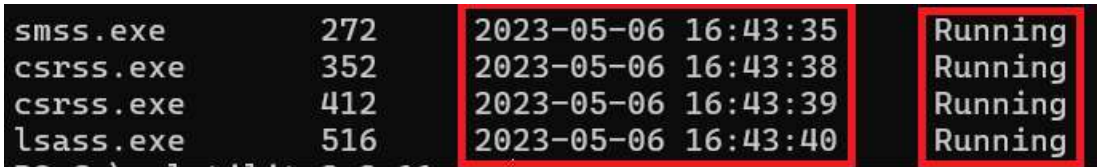
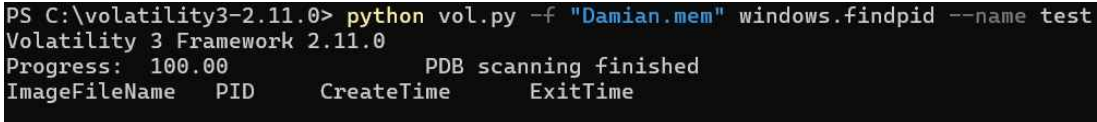
[그림 1-2] findpid 플러그인 설명 출력

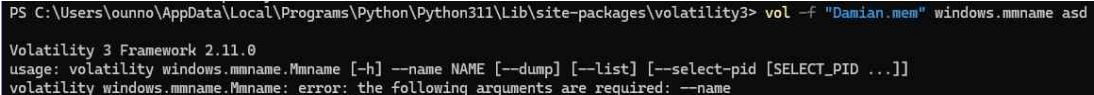
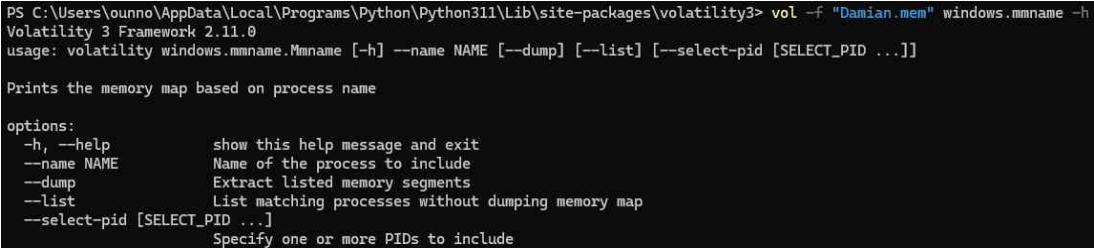
```
windows.mmname.Memmap
Prints the memory map using either --pid or --name (exact match on
```

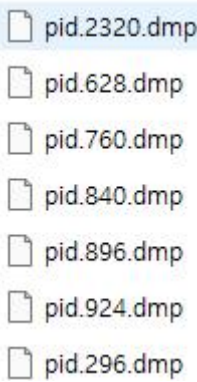
[그림 1-3] mmname 플러그인 설명 출력

테스트 번호	테스트 종류	테스트 항목	담당자
121	단위 테스트	get_requirements(cls) 단위 테스트	김민서
함수 명			
def get_requirements(cls):			
테스트 목적			
플러그인에서 요구하는 구성 요소(requirements)가 Volatility3의 실행 요구사항에 맞게 정확히 정의되었는지 확인			
테스트 환경			
OS: Window11			
.mem file: Damian.mem			
Plugin: findpid.py			
테스트 케이스			
<ol style="list-style-type: none"> 1. ModuleRequirement 인식 확인 : kernel 요구사항 포함 2. StringRequirement 인식 확인 : --name 인자 없이 실행 시 필수 요구 항목 누락 오류 발생 3. BooleanRequirement 동작 확인 : --exact 옵션은 기본값 False이며 생략 가능해야 함 4. python vol.py findpid-h 시 옵션 설명 확인 : 해당 옵션들에 대한 설명이 올바르게 출력되어야 함 			
테스트 결과			
모든 요구 사항이 올바르게 정의됨.			
			
[그림 2-1] 필수 옵션 누락 시 오류 메시지 출력			
특히 --name 필수 옵션이 누락될 경우 Volatility3 프레임워크에서 적절히 오류를 발생시켰으며, --exact 옵션은 선택적으로 처리됨.			
			
[그림 2-2] help 명령어 입력 시 설명 메시지 출력			
명령어 도움말에서도 각 인자가 설명과 함께 명확히 출력됨.			

테스트 번호	테스트 종류	테스트 항목	담당자
122	단위 테스트	def _generator(self) -> Iterator: 단위 테스트	김민서
함수 명			
def _generator(self) -> Iterator:			
테스트 목적			
프로세스를 순회하면서 --name 및 --exact 조건에 따라 정확한 필터링이 수행되는지, 시간 정보가 정상 출력되는지 확인			
테스트 환경			
OS: Window11			
.mem file: Damian.mem			
Plugin: findpid.py			
테스트 케이스			
5. --name ss 실행 시 결과 포함 : 'ss' 문자열을 포함하는 프로세스들이 정상 출력되어야 함 6. --name lsass.exe --exact 실행 시 정확 일치 : smss.exe 등은 제외되고 lsass.exe 만 출력되어야 함 7. create_time, exit_time 값 확인 : 종료되지 않은 프로세스는 Running, 그 외는 종료 시간 출력			
테스트 결과			
 <pre> PS C:\volatility3-2.11.0> python vol.py -f "Damian.mem" windows.findpid --name ss Volatility 3 Framework 2.11.0 Progress: 100.00 PDB scanning finished ImageFileName PID CreateTime ExitTime smss.exe 272 2023-05-06 16:43:35 Running csrss.exe 352 2023-05-06 16:43:38 Running csrss.exe 412 2023-05-06 16:43:39 Running lsass.exe 516 2023-05-06 16:43:40 Running </pre>			
[그림 3-1] 'ss' 문자열 포함 프로세스 출력			
 <pre> PS C:\volatility3-2.11.0> python vol.py -f "Damian.mem" windows.findpid --name lsass.exe --exact Volatility 3 Framework 2.11.0 Progress: 100.00 PDB scanning finished ImageFileName PID CreateTime ExitTime lsass.exe 516 2023-05-06 16:43:40 Running </pre>			
[그림 3-2] '--exact' 옵션 적용 후 프로세스 출력			
--name에 대한 부분 포함 검색과 --exact에 대한 정확 일치 조건 모두 정확히 동작함. 시간 값은 datetime 형식으로 포맷된 문자열로 출력되며, 종료되지 않은 프로세스에 대해서는 "Running"으로 표기됨.			

테스트 번호	테스트 종류	테스트 항목	담당자
123	단위 테스트	run(self) 단위 테스트	김민서
함수 명			
def run(self):			
테스트 목적			
_generator()에서 수집한 데이터를 TreeGrid 형식으로 변환하여 Volatility3 프레임워크에서 정상 출력하는지 확인			
테스트 환경			
OS: Window11			
.mem file: Damian.mem			
Plugin: findpid.py			
테스트 케이스			
8. TreeGrid 컬럼 구조 확인 : 출력 결과가 ImageFileName, PID, CreateTime, ExitTime의 컬럼 구조를 가짐			
9. 결과 값 확인 : 필터 조건에 맞는 프로세스만 정확히 출력됨			
10. 결과 없음 처리 : 존재하지 않는 프로세스 이름 지정 시 에러 없이 으로 정상 종료됨			
11. 대량 프로세스 출력 시 성능 확인 : 수십 개 프로세스를 출력해도 오류 없이 렌더링 됨			
테스트 결과			
			
[그림 4-1] TreeGrid 컬럼 구조 확인			
			
[그림 4-2] 존재하지 않는 프로세스 지정 시 에러 없이 정상 종료			
<p>TreeGrid 포맷이 적절하게 적용되었으며, 모든 컬럼이 정상 출력됨.</p> <p>데이터 수가 많아도 지연이나 예외 없이 전체 결과가 표시되었고, 필터링 조건이 없거나 불일치할 경우에도 잘 처리됨.</p>			

테스트 번호	테스트 종류	테스트 항목	담당자
131	단위 테스트	사용자에게 요구되는 필수 및 선택 인자 정의 정확성 확인	최원혁
함수 명			
def get_requirements(cls) -> List[interfaces.configuration.RequirementInterface]:			
테스트 목적			
Volatility3의 실행 요구사항에 맞게 요구 인자 정의 정확성 확인 (--name, --list, --dump, --select-pid) 및 명령어 도움말 출력 여부 확인			
테스트 환경			
OS: Window11			
.mem file: Damian.mem			
Plugin: mmname.py			
테스트 케이스			
<ol style="list-style-type: none"> 1. ModuleRequirement 인식 확인: kernel 요구사항 포함 2. PluginRequirement 인식 확인: 플러그인 실행가능 버전 및 참조 플러그인 확인 3. StringRequirement 인식 확인: 문자열 입력을 요구, --name 인자 없이 실행 시 필수 요구 항목 누락 오류 발생 4. BooleanRequirement 작동 확인: 기본값 False, 옵션 선택 여부 가능 5. ListRequirement 작동 확인: 여러 PID를 수신 여부 6. help 명령어 작동 및 설명 확인 : 옵션들에 대한 설명 여부 및 작동 확인 			
테스트 결과			
모든 요구 인자가 올바르게 정의됨.			
			
[그림 5-1] 필수 옵션 누락 시 오류 메세지 출력			
<p>--name 필수 옵션이 누락될 경우 Volatility3 프레임워크에서 오류 발생. 기타 옵션들은 선택적으로 처리됨.</p>			
			
[그림 5-2] help 명령어 입력 시 설명 메시지 출력			
명령어 도움말에서도 각 인자가 설명과 함께 명확히 출력됨.			

테스트 번호	테스트 종류	테스트 항목	담당자
132	단위 테스트	메모리 매핑 정보 출력 및 덤프파일 생성 테스트 (->TreeGrid 형태로 생성)	최원혁
함수 명			
def _generator(self, procs):			
테스트 목적			
메모리 매핑 정보 출력(TreeGrid 형태) 여부 (+ 내용 정확성 확인), contextlib.ExitStack() 사용 with 문 안에서 파일 핸들링 -> pid.XXXX.dmp 형식 준수 여부			
테스트 환경			
OS: Window11			
.mem file: Damian.mem			
Plugin: mmname.py			
테스트 케이스			
7. 메모리 매핑 출력 결과 확인: 프로세스에 대해 가상/물리 주소, 크기, 오프셋, 파일 출력 정보 출력			
8. 다수 메모리 매핑 출력 결과 확인: 동명의 프로세스가 다수 일 때, 전 목록 출력			
9. 단일 dump 여부 확인: 메모리 구간을 .dmp 파일로 저장			
10. 다수 dump 여부 확인: 다수의 메모리 구간을 .dmp 파일로 저장			
11. .dmp 형식 준수 확인: 덤프를 수행할 경우 pid.XXXX.dmp 형식의 파일 생성			
테스트 결과			
모든 테스트 케이스 정상 작동 확인.			
<pre> 0xfa800ac00000 0x118a00000 0x200000 0x1f502000 Disabled 0xfa800ae00000 0x118800000 0x200000 0x1f702000 Disabled 0xffffffff00000 0x1000000 0x1000 0x1f902000 Disabled 0xffffffff01000 0x1050000 0x1000 0x1f903000 Disabled 0xffffffff02000 0x1010000 0x4000 0x1f904000 Disabled 0xffffffff06000 0xfec00000 0x1000 0x1f908000 Disabled 0xffffffff07000 0x5000 0x1000 0x1f909000 Disabled 0xffffffff08000 0x1000 0x1000 0x1f90a000 Disabled 0xffffffff09000 0x6000 0x1000 0x1f90b000 Disabled 0xffffffff0a000 0x0 0x1000 0x1f90c000 Disabled 0xffffffff0b000 0x7000 0x1000 0x1f90d000 Disabled 0xffffffff0c000 0x2000 0x3000 0x1f90e000 Disabled 0xffffffffe0000 0xfec00000 0x1000 0x1f911000 Disabled PS C:\Users\ounno\AppData\Local\Programs\Python\Python311\Lib\site-packages\volatility3> vol -f "Damian.mem" windows.mmname --name notepad.exe </pre>			
[그림 6-1] 메모리 매핑 출력 결과			
			
[그림 6-2] 다수 dump 여부 결과			

테스트 번호	테스트 종류	테스트 항목	담당자
133	단위 테스트	프로세스 필터링, 옵션 조합 테스트, 예외 메시지 출력 및 정확성, 일관성 확인(실행 로직, 제어흐름 분기 확인)	최원혁
함수 명			
def run(self):			
테스트 목적			
실행 로직, 제어흐름 분기 확인, 예외처리 확인 (예외 메시지 출력, 디코딩 실패, Timestamp 파싱 실패 등의 예외는 개별 try-except로 처리) 작동 방식 확인(프로세스 필터링, _generator 호출 여부, PID 선택 처리, 리스트 출력)			
테스트 환경			
OS: Window11			
.mem file: Damian.mem			
Plugin: mmname.py			
테스트 케이스			
12. 리스트 출력 결과 확인: 매칭된 프로세스들의 이름, PID, 생성 시각, 종료 여부를 포맷에 맞춰 출력, 여러 프로세스가 있는 경우 --select-pid를 안내			
13. PID 선택 처리 확인: --select-pid 옵션 시 그에 해당하는 프로세스만 추림, --dump 없이 여러 PID가 선택된 경우 → 에러 메시지 출력			
14. 예외 처리 확인: 이름이 없거나 매칭된 프로세스가 없을 경우 오류 메시지 출력			
테스트 결과			
모든 테스트 케이스 정상 작동 확인.			
<pre> PS C:\Users\ounno\AppData\Local\Programs\Python\Python311\Lib\site-packages\volatility3> vol -f "Damian.mem" windows.mmname --name svchost.exe --list Volatility 3 Framework 2.11.0 Progress: 100.00 PDB scanning finished Message [mmname] Matching processes with name 'svchost.exe': ----- ImageFileName PID CreateTime ExitTime ----- svchost.exe 628 2023-05-06 16:43:40 Running svchost.exe 760 2023-05-06 16:43:41 Running svchost.exe 840 2023-05-06 16:43:41 Running svchost.exe 896 2023-05-06 16:43:41 Running svchost.exe 924 2023-05-06 16:43:41 Running svchost.exe 296 2023-05-06 16:43:41 Running svchost.exe 348 2023-05-06 16:43:41 Running svchost.exe 1180 2023-05-06 16:43:42 Running svchost.exe 2168 2023-05-06 16:45:49 Running ----- --> If there are multiple processes, Please re-run with --select-pid <PID> to proceed. </pre>			
[그림 7-1] 리스트 출력 결과			
<pre> PS C:\Users\ounno\AppData\Local\Programs\Python\Python311\Lib\site-packages\volatility3> vol -f "Damian.mem" windows.mmname --name svchost.exe --select-pid 628 760 Volatility 3 Framework 2.11.0 Progress: 100.00 PDB scanning finished Message --> [mmname] Only one process memory map can be printed. Please select only one process. </pre>			
[그림 7-2] PID 선택 처리 결과, dump 없이 여러 PID가 선택 시 에러 메시지 결과 출력			
<pre> PS C:\Users\ounno\AppData\Local\Programs\Python\Python311\Lib\site-packages\volatility3> vol -f "Damian.mem" windows.mmname --name noname.exe Volatility 3 Framework 2.11.0 WARNING volatility3.plugins.windows.mmname: [mmname] No processes found matching name: noname.exe Message [mmname] No processes found matching name: noname.exe </pre>			
[그림 7-3] 예외 처리 확인 결과, 이름이 없는 프로세스 검색 시 결과 출력			

② 결함

- 총 버그 수: 해당사항 없음
- 버그 상태(Open, Closed, Responding): 해당사항 없음
- 버그 수(Open, Resolved, Closed): 해당사항 없음
- 심각도 및 우선순위 구분: 해당사항 없음

③ 특이사항

- TC에서 미진행한 케이스 및 사유: 해당사항 없음
- 아직 해결되지 않은 이슈 및 사유: 해당사항 없음