리버싱 이 정도는 알아야지

Intro. 리버싱 시작하기

http://securityfactory.tistory.com

목 차

Intro

리버싱 시작하기

SECURITY FACTORY

1. 분석환경 만들기

1.1. 컴퓨터 구성의 이해

간단하게 Sample 01.exe 를 분석해보려고 합니다. 그런데 분석을 하려면 대략적으로 컴퓨터의 구성이 어떻게 되는지 알아야 합니다. CPU, 메인 메모리, 하드디스크는 컴퓨터를 구성하는 가장 기본이면서 핵심이 되는 장치입니다.

그림 1-1 컴퓨터를 구성하는 핵심 장치

이들 장치들의 개념은 다음과 같습니다.

용 어	정 의
CPU	CPU는 'Central Processing Unit'의 약자로서, 직역하면 중앙처리장치다. 단어 그대로, 컴퓨터의 중앙에서 모든 데이터를 연산 및 처리한다.
메인 메모리(RAM)	주기억장치는 현재 실행 중에 있는 프로그램과 이 프로그램이 필요로 하는 데이터를 일시적으로 저장하는 장치다.
하드 디스크	하드 디스크는 컴퓨터 본체를 구성하는 부품 중 하나로서 필요한 자료를 저장하는, 저장 공간으로서의 역할을 한다.

표 1-1 핵심 장치 개념 정보

이렇게 봐선 무슨 말인지, 어떻게 활용되는지 잘 모르겠네요. 용어의 사전적인 개념은 단번에 이해하기 어렵습니다. 그렇기 때문에 개념을 외우기 보다 나만의 언어로 이해하는 것이 중요합니다. 사실 IT는 우리가 사는 세상과 많이 닮아있습니다. 적절한 예시와 비유를 들어서 한번 살펴보죠.

전원주택을 지으려고 합니다. 침실, 화장실, 놀이방, 작업실 정도면 충분할거 같습니다. 이대로 건축 사무소에 의뢰하니 건물이 만들어졌습니다. 그런데 아직 사람이 살기엔 부족한 부분이 많네요. 도배부터 시작해서 전자기기와 가구를 들이는 등의 내부 인테리어 작업까지 완료해야 비로소 사람이 살 수 있는 집

이 됩니다.

이번엔 컴퓨터를 맞춰봅시다. 먼저 스펙과 견적을 고려해서 장치들을 선택합니다. 여기서 장치는 메인보드, CPU, 메인 메모리, 하드디스크 정도가 될 것입니다. 적절한 사양을 결정하고 조립하면 본체가 만들어집니다. 그리고 여기에 운영체제를 설치해야 비로소 우리가 쓸 수 있는 컴퓨터가 완성됩니다.

둘이 많이 닮아있지 않나요? 컴퓨터는 실행파일(프로그램)이 거주하는 집과 같습니다. 부지가 되는 메인보드 위에 실행파일이 일하는 공간(메인 메모리)과 저장 공간(하드 디스크), 중앙처리장치(CPU)를 갖춰 놓고, 인테리어 작업(운영체제 설치)까지 마무리하면 실행파일이 동작할 수 있는 환경이 됩니다. 이때부터 우리는 마우스를 사용해서 실행파일이 일을 하거나 쉬게 만들 수 있습니다. 이렇게 보니 상당히 단순하네요. 컴퓨터가 복잡하고 어려워 보이지만 원리를 파악하고 나면 그렇지도 않습니다.

1.2. **분석환경 선택 및 설정**

① 운영체제 설치

다양한 환경에서 많은 시도를 해보는 것은 매우 좋은 경험입니다. 그러나 처음 공부하는 입장에서는 돌발 변수를 최소화 하는 것이 좋습니다. 앞서 언급한 것과 같이『리버싱 이 정도는 알아야지』는 Windows 7 운영체제 32비트 환경을 기준으로 작성되었습니다. 동일한 환경에서 실습하기 바랍니다.

보통 분석 작업은 가상환경에서 진행합니다. 'VMware 윈도우7 가상환경 만들기, VMware에 윈도우 설치하기, VirtualBox Windows 설치…' 와 같은 키워드로 검색하면 쉽게 환경을 만들 수 있습니다. 가상환경에서 분석하는 습관을 들이기 바랍니다.

여기에 몇 가지 분석 도구를 설치하겠습니다. 최소한으로 필요한 도구만 사용하려고 합니다. 지금 우리에겐 최소한의 정보만 가지고, 직접 찾고 알아내면서 익히는 과정이 필요합니다.

※ 참고

"분석 환경을 세팅하세요."라고 던져놓기만 하면 너무 무책임해 보이죠? 정확한 가이드가 없으면, 이게 맞는건지 의구심이 들기도 할겁니다. 그런데 세부적인 설정에 신경 쓰지 마세요. 그냥 웹에서 설명하는 기본 설정에 따르면 됩니다. 그냥 설치합시다!!

② 동적 분석 도구 세팅

우리가 동적 분석 도구를 사용하는 이유는 본격적인 분석에 앞서 사전 정보를 얻기 위함입니다. 그리고 그 목적에 적당한 도구를 사용할 겁니다. 크게 프로세스 동작 확인, 파일 변화 확인, 네트워크 행위 확인 3 가지 입니다.

도 구	설 명
Procexp.exe	프로세스와 관련된 다양한 정보들을 제공해준다.
FileMonitor.exe	파일 시스템에서 일어나는 이벤트(파일 생성, 수정, 삭제)를 모니터링 해준다.
Tcpview.exe	TCP와 UDP 등 일련의 네트워크 연결 상태를 제공해준다.

표 1-2 동적 분석 도구 소개

Procexp.exe 는 프로세스 모니터링 도구입니다. 이 도구를 사용하면 컴퓨터에서 동작하고 있는 프로세스들의 정보들(프로세스 상태, Thread 정보, PID, 모듈 리스트 등)을 실시간으로 확인할 수 있습니다. 과유불급이라는 사자성어가 있죠? 실로 Procexp.exe 는 프로세스에 대한 매우 많은 정보를 제공해줍니다. 그러나 우리는 샘플 파일이 잘 동작하는지 확인하는 용도가 될 겁니다. 그 외에는 필요할 때마다 그때 그때 활용하겠습니다.

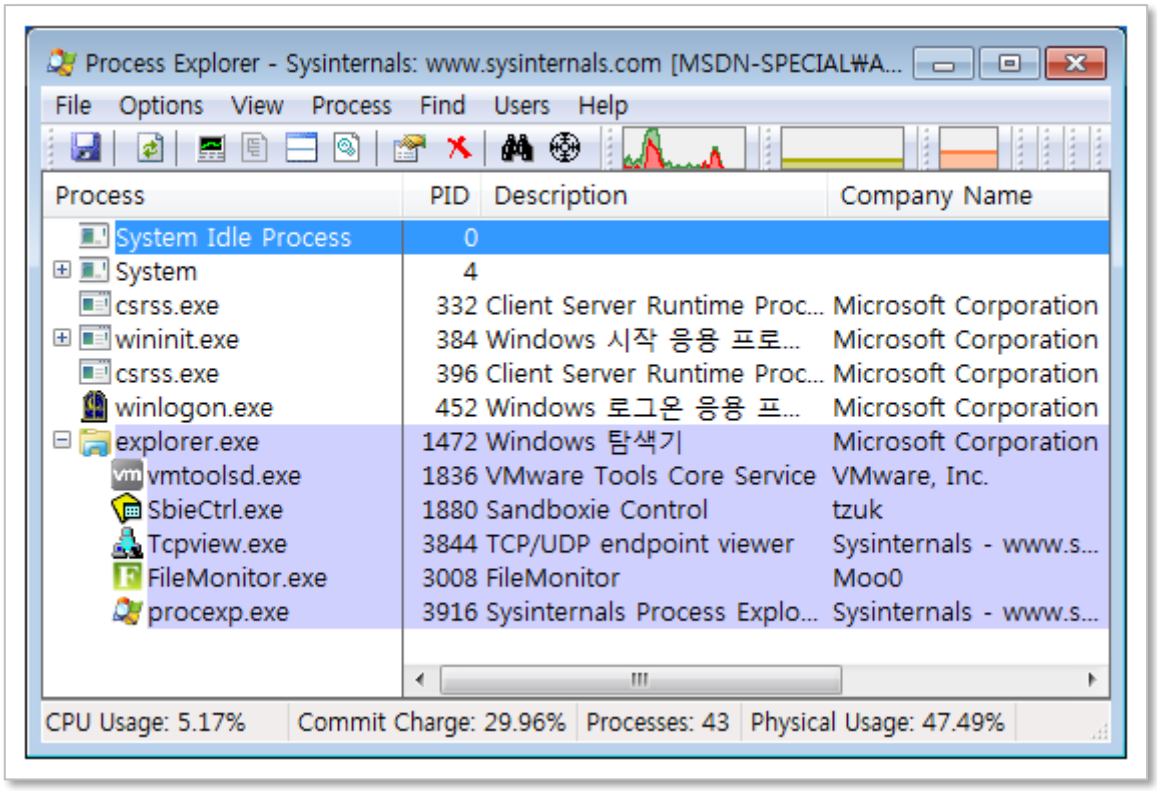

그림 1-2 Procexp.exe

FileMonitor.exe 는 Moo0(https://kor.moo0.com/)에서 제공하는 파일 모니터링 도구입니다. 프로세스가 파일 시스템에 존재하는 파일들을 수정, 삭제하거나 새로 생성할 때, 행위 정보들을 확인할 수 있습니다. 다만 그 대상이 시스템 프로세스를 포함한 모든 프로세스이기 때문에 내가 분석하는 프로세스의 행위 정보를 잘 걸러서 볼 줄 알아야 합니다.

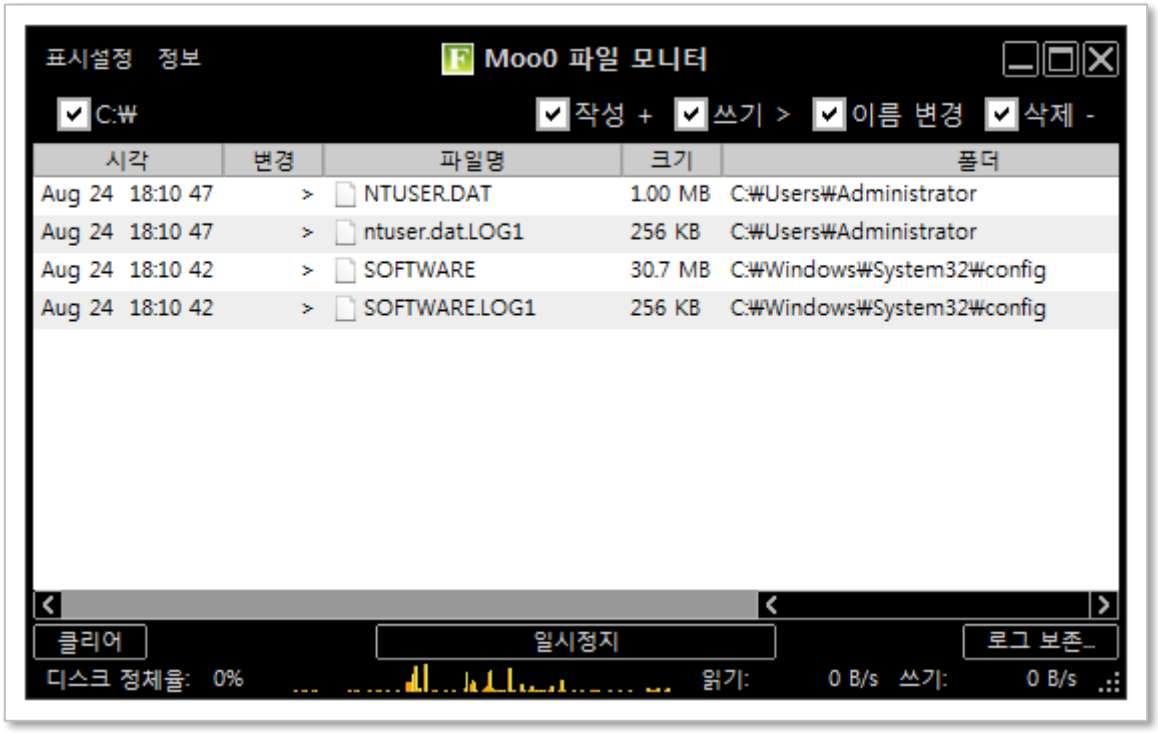

그림 1-3 FileMonitor.exe

TCPView.exe 를 사용하면 네트워크 연결 상태를 확인할 수 있습니다.

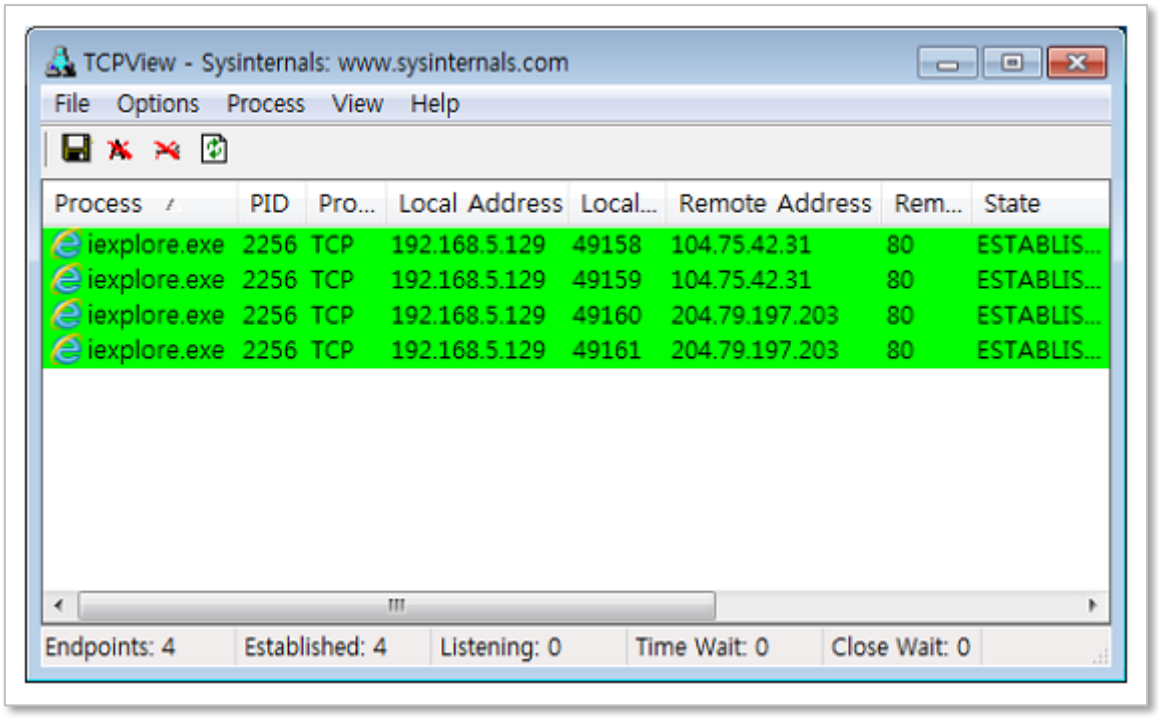

그림 1-4 TCPView.exe

동적 분석 도구를 선택했으니 잘 활용해야 합니다. 정해진 방법은 없지만 필자는 흥미 있는 파일이 생기거나 분석을 해야 하는 상황이 오면, 동적 분석 도구들을 바탕화면에 띄워 놓고 한번 실행해 봅니다. 그리고 파일이 어떻게 동작하는지 모니터링 합니다. 이렇게 하면 분석이 필요한 파일인지 파악할 수 있고, 그에 필요한 사전 정보도 얻을 수 있습니다.

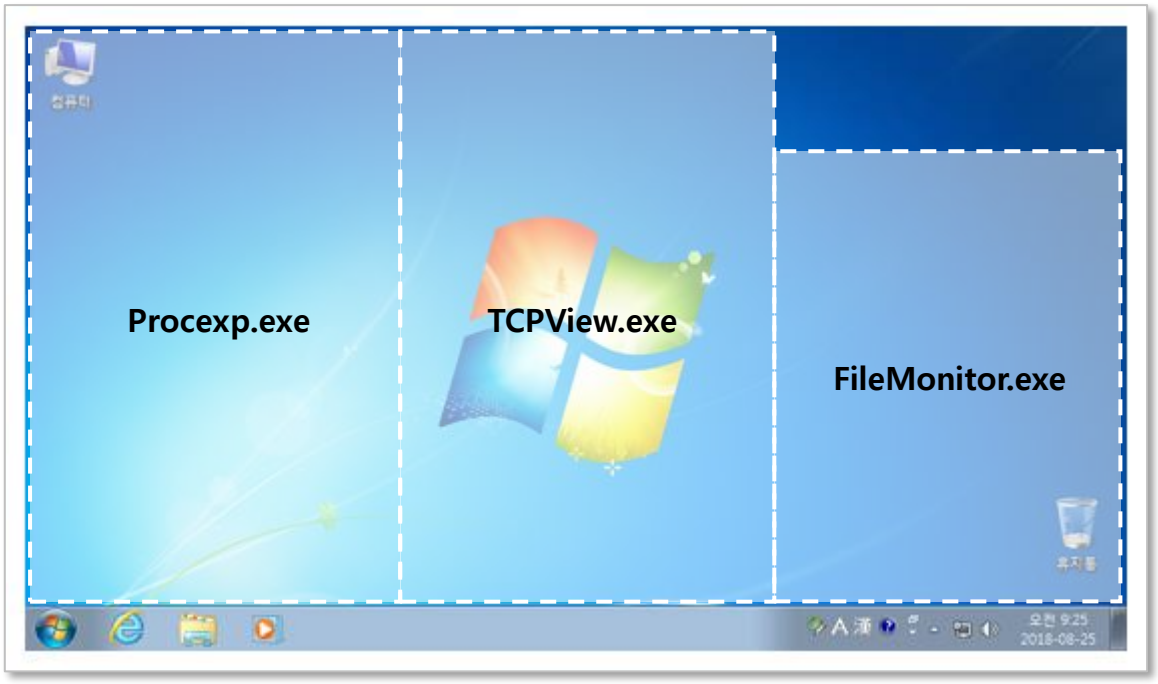

그림 1-5 분석 환경 기본 구성

다음은 동적 분석 도구를 다운로드 받을 수 있는 경로입니다. 검색엔진에서 파일이름으로 검색해도 나옵니다.

도 구	다운로드 경로
Procexp.exe	https://live.sysinternals.com/procexp.exe
Tcpview.exe	https://live.sysinternals.com/tcpview.exe
FileMonitor.exe	http://www.moo0.com/software/FileMonitor/download/free/

표 1-3 동적 분석 도구 다운로드 경로

③ 디버거 선택

OllyDBG, WinDBG, Immunity Debugger, x64 DBG 등 다양한 디버깅 도구들이 있습니다. 그 중에서 OllyDBG.exe 를 사용하겠습니다. 각 도구들 마다 강점이 있지만 기본 동작 원리는 크게 다르지 않습니다. 그렇기 때문에 의미를 두지 않았으면 합니다. 시간이 지나면 자연스럽게 모든 도구를 사용해보게 될 겁니다. 그리고 그 과정에서 도구를 선택하는 기준이 생기리라 생각합니다. 참고로 필자는 서비스 분석, 커널 분석, 64 비트 파일 분석에 따라 적절한 디버거를 선택해서 사용합니다.

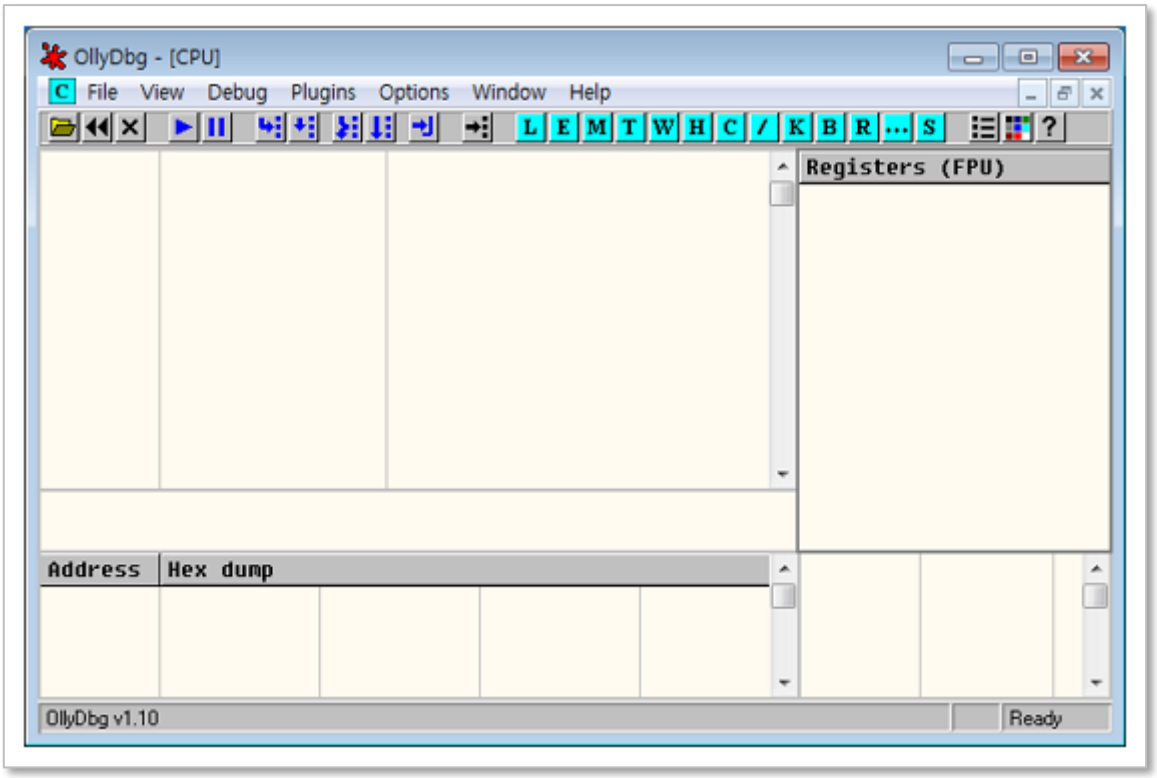

그림 1-6 OllyDBG.exe

분석 도구의 사용법은 알아야겠죠? 다음은 유용한 단축키 정보입니다. 이 외에도 많은 단축키가 있지만 많이 사용하는 것과 필요한 것 위주로 작성했습니다. OllyDBG 사용에 있어 이 정도만 알아도 충분합니다.

OllyDBG.exe 사용이 처음이라면 실행 파일을 올려놓고 단축키를 하나씩 사용해보면서 손에 익히시기 바랍니다. 단축키는 많이 사용해서 익숙해지는 것이 좋습니다. (실습간 수시로 참고하세요.)

단축키 정보	설 명
F2	- BreakPoint를 설치하고 해제한다.
F7	- 하나의 명령어를 실행한다. - Call 명령어 실행 시 해당 함수 내부로 들어간다.
F8	- 하나의 명령어를 실행한다. - Call 명령어 실행 시 해당 함수 내부로 들어가지 않는다.
F9	- 실행 (Excute)
Ctrl + F2	- 디버깅을 처음부터 다시 시작한다. (재실행)
Ctrl + F7	- Step Into 명령어를 반복 실행한다.
Ctrl + F8	- Step Over 명령어를 반복 실행한다.
Ctrl + F9	- 해당 함수 내에서 RETN 명령어까지 실행한다.
Ctrl + G	- 원하는 주소로 이동한다.

Ctrl + E	- 데이터를 수정 화면을 연다.
Alt + E	- 로드되어 있는 모듈 리스트를 확인한다.
Alt + M	- Memory Map을 확인한다.
Alt + C	- 실행 명령 위치로 돌아간다.
-	- 커서가 - 위치로 이동한다.
+	- 커서가 + 위치로 이동한다.
Enter	- 커서가 Call이나 JMP 명령어에 위치해 있으면 해당 주소를 따라가서 보여준다. (실행되는 것은 아니다.)

표 1-4 OllyDBG 단축키 정보

※ OllyDGB 다운로드 경로:
https://drive.google.com/file/d/1cM1mqZ2xnA1ExApGThmjYTcCabGDYl2s/view?usp=sharing

2. 공부 방향에 대한 설명

공부를 시작하는 사람들을 보면 의지가 불타오릅니다. 그만큼 간절하고 각오가 남다른 거겠죠. 이런 마음 때문일까요? 분석을 하다 보면 왠지 다 알고 넘어가야 할 것 같습니다. 그러지 않으면 뭔가 찜찜하면서 뒤쳐진 느낌입니다. 저도 그랬습니다.

개인적으로 가장 안 좋은 습관 중에 하나가 '한번에 모든걸 알아야 한다.'는 강박관념이라고 생각합니다. 누구나 알고자 하는 욕심이 있고 많은 것을 알고 싶어하지만, 모두가 이를 만족하진 못하죠. 이는 자칫 흥미를 잃고 포기하는 결과를 초래하기도 합니다.

리버싱은 아주 긴 마라톤과 같습니다. 그렇기 때문에 남들보다 조금 빨리 알았다고 해서 우쭐댈 필요 없고, 조금 늦었다고 위축될 필요도 없습니다. 답을 빨리 찾는 것이 능사가 아닙니다.

철수와 영희에게 신촌에서 동서울터미널을 찾아가는 미션이 주어졌다고 칩시다. 철수는 친구가 알려줘서 바로 갈 수 있었지만, 영희는 길을 몰라서 동대문, 사당, 강남을 헤매다가 결국 찾아가지 못했습니다. 여기서 승자는 철수인가요? 미션이 여기서 끝이라면 승자는 당연히 철수 입니다. 그런데 다음 미션이 동대문 찾아가기라면 얘기가 달라지겠죠. 동대문, 사당, 강남이 아닌 제 3의 장소라도 마찬가지 일겁니다. 적어도 영희는 동서울터미널을 찾아가기 위해 시행착오를 겪었기 때문에 철수보다는 길을 빨리 찾으리라 생각합니다.

파일 분석도 마찬가지 입니다. 예를 들어보죠.

	File 01	File 02	File 03	File 04
Function A	☑	☑	☑	☑
Function B	☑		☑	
Function C	☑	☑		☑
Function D				☑
Function E		☑	☑	☑

그림 2-1 각 파일의 기능 정보

File 01은 A, B, C 세 가지 기능을 가지고 있습니다. 이 것을 분석해야 합니다. 그런데 너무 어렵네요. 각각의 기능을 분석하는데 오랜 시간과 노력이 들었습니다. 그 결과 기능 A의 동작과 원리는 완벽하게 파악했지만, B, C는 동작을 확인하는데 그쳤습니다.

다음으로 File 02를 분석하게 되었습니다. File 02는 A, C, E 기능을 가지고 있습니다. 다행히도 기능 A는 완벽하게 알고 있습니다. 기능 C도 File 01을 분석하면서 많이 봤기 때문에 익숙합니다. 거기다가 시간이 지나고 여유를 좀 가졌더니, 안보이던 것들이 보이면서 해결 방법이 떠올랐습니다. File 01에 비해 적은 시간과 노력으로 A, C 기능을 분석했습니다. 그래서 남은 시간에 기능 E를 분석하는데 온전히 집중할 수 있었습니다.

누차 강조하지만 우리는 공부하는 사람입니다. 결과에 집착하지 마세요. 다양한 경우를 염두 해두고 그 안에서 답을 찾아가는 과정을 즐기기 바랍니다. 말도 안 되는 상상의 나래를 펼쳐도 좋습니다. 설령 답을 찾지 못해도 상관없습니다. 이 모든 시행착오가 밑거름이 되어서 역량을 키우는 날개가 되어 줄 것입니다.

그리하여 『리버싱 이 정도는 알아야지』는 독자들이 자신의 수준에 맞게 공부할 수 있도록 구성했습니다. 목차를 살펴보세요. 분석이 단계 별로 나누어져 있죠? 여기서 모든 단계를 다 이해하고 넘어갈 필요는 없습니다. 샘플 파일을 실행해보는 것으로 끝나더라도 그 과정과 원리를 고민했다면 충분한 가치가 있습니다.

Chapter 01 무작정 분석해보기

1. 실행파일 동작 확인
2. 코드 분석_Level.1 | 흐름 파악하기
3. 코드 분석_Level.2 | API 호출 분석
4. 코드 분석_Level.3 | 파고들기
5. 코드 구현_Level.4

리버싱 이 정도는 알아야지

발행일 | 2018 년 09 월
발행자 | SecurityFactory
페북 주소 | http://bitly.kr/OrHQ
이메일 | itseeyou@naver.com


SECURITY FACTORY
http://securityfactory.tistory.com