킥오프 이후 첫 멘토링

그주에 있었던 본인들에 대한 성과 이야기하고 다음시간은 다음주에 할 일을 이야기하고자 합니다.

한주동안 진행했던 일 이야기

김성민의 피드백

* 메크로 방식이 무엇인지?
* Remote template injection이 무엇인지?  
  관련 링크는 다음과 같음   
  <https://blog.sunggwanchoi.com/remote-template-injection/>   
  해당 단어가 생소하다.  
  여러가지 문서의 유향이 있음  
  문서의 유형 중 새로운 유형인거와 같음  
    
  성과로써 리모트 템플릿을 말하자.  
    
  난독화가 어디가 문제냐?  
  드롭하고 떨어지는 파일에 난독화가 너무 심하게 걸려있다.  
  파일은 실행인지 잘 모르겠음.  
  매크로로 인해 생성하는 파일이 어떤 것인지 모른다.  
  파일은 문서, 실행파일, 스크립트가 있음.  
  다음에는 상황을 보여달라.

김수종 관련 피드백

* 가상화 관련해서 논문을 찾아 봤음.  
  이중에서 눈에 띄는 것은 문서형 악성코드 피해 방지를 위한 문서 격리 시스템을 중점적으로 봤음.  
  멘토님이 알려주신 가상화 관련해서 개념이 없어 찾아보기로 했다.  
  논문에서 doc sandbox에 대해 설명을 했음.  
  해당 논문이 멘토님이 이야기한 어플리케이션 가상화와 다르다.  
  따라서 멘토님이 생각하신 가상화가 어떤 가상화 인지 알고 싶다.  
  문서 자체만 격리인지?   
  원래 있던 개념인지? 궁금했다.  
  논문에 대한 정리는?   
  🡪 저희가 생각했던 논문과 다른 방향인 것 같다.  
    
  다른 논문에서는 어플리케이션 가상화를 하는 것 같아 봤는데 어플리케이션 분할기술에서 사용되는 기술이 비슷한 것 같았다.  
    
  위키피디아 보면서 존재하는 기술들에 대해 찾아봤다.  
  해당 기술들의 문서 찾아보면서 읽었는데 방향이 같은 지 안와다았다.  
  따라서 멘토님이 이야기하신 가상화가 무엇인지 알고 싶다.  
  🡪 아이디어만 주고 방향을 찾는 것은 우리다.  
  수종이가 찾은 논문들은 성과로 사용하자.  
  이런 생각을 찾아 이런 생각을 했다라고 이야기해야 했다.  
  논문을 읽고 이러한 개념을 잡았다가 나와야 한다.  
  방향이 잘못됬음.  
  🡪 정답이 맞는지 확인했으니 달랐다.  
  🡪 일주일 동안 성과가 나오지 않았다. 🡺 가상화 관련해서 개념이 잡히지 않아서.  
  논문 주제는 나쁘지 않음 우리들끼리 이해해야 한다.

이예찬 관련 피드백

* 주로 조사한것은 사용자가 파일을 실행할 때 어떻게 인지하고 파일을 전달할지에 대해 찾아 봤다.  
  온디멘드와 온엑세스 방식이 있음  
  온디멘드는 직접 사용자가 바이러스 토탈처럼 업로드하는 것.  
  온엑세스는 자동으로 시스템 탐지가 온엑세스이다.  
    
  프로토 타입으로 만들 수 있는게 온디멘드 방식이라 온디멘드 방식으로 개발하고 실제 상용백신처럼 사용자가 파일을 실행할 때 감지하는 것을 목표로한다.  
  이를 위해 자료를 찾아보니   
  단순히 유저 분야에서는 좁은 범위의 정보를 찾을 수 있고 모든 프로세스를 검사하고 악성 프로세스는 실행되어있기에 늦었다 생각함.  
  파일 필터의 미니 필터 드라이브를 찾았다.  
  이를 바탕으로 진행하는 것을 목표로 잡았다.

정경재 관련 피드백

* 우크라이나 전쟁때 사용되었던 악성코드이다라고 말하자  
  뒤집어질 내용을 찾아서 말해주자.  
    
  진단 샘플도 준비해서 보여주자.

정현구 관련 피드백

* 리버싱 관련해서 기초를 알아야 한다 생각해서 IDA관련 사용법 공부.
* 주간 보고서 작성 마무리 작성을 했다.
* 깃허브에 어플리케이션 가상화라는 키워드를 통해서 검색했을 때 가장 우리의 프로젝트와 맞는 프로젝트가 있어서 분석하는 것을 진행  
  어떤 목적성을 가지고 소스를 받지 않았나?  
  🡪 가상화에 대한 정보를 몰라 깃허브에 검색해 코드를 받음.

🡪 전상현 멘토님  
첫주치고는 성과가 많이 나오지 않았다.

멘토님이 보시기에 아쉬운 모습이 보인다.  
이렇게 진행하면 누군가 심히 고생해야 한다.

이야기 말할 때 자격이 있어야한다.

🡪 남성엽 멘토님

취약점 관련해서 이야기하면 문서 관련해서 offensive를 만들때 취약점을 만들때 2가지로 접근

가장 선호하는 문서를 접근해서 악성코드를 만듬

가장 많이 고려한것은 폰트이다.

폰트 취약점을 찾으면 다양한 문서에서 사용이 가능하다.

문서 자체 안으로 시작하지만 확장해서 폰트 취약점을 찾는 것도 하면 가장 좋은 방어 일듯하다. 🡪 고려하자.

최근의 트랜드는 메크로 기반으로 많이 한다.

취약점이 나오기 어려워지면서 apt팀이 쉬운쪽으로 간다.

시간적 여유가 된다면 폰트도 추가하면 좋을 듯하다.

해당 부분도 찾아보자.

앱의 취약점도 좋지만 다른 취약점도 찾아보자.

조금 더 열심이 하자.

전상현 멘토님

기획서나 설계도 이야기가 없음.

킥오프 이후에 변경되었기에 문서가 필요하다.

문서 공유로는 끝나서 안된다.

너무 심플하다.

아직 기획 설계 코딩 테스트 배포 이다.

기획과 설계가 안되어 힘듯하다.

김성민의 질문

프로세스 인젝션을 이야기했었는데 김종민 멘토님이 시켜서 했어서 근본없이 공부했다.

악성코드 공부할 때 한계를 느낀다.

악성코드 분석에 도움이 되는 내용이나 로드맵 등 서적이나, 다른 방법이 있나?

🡪 엔진 개발은 분석과 양상이 다르다.

엔진 개발자가 분석된 것을 통해 개발하니 레지스터리나 디버거의 브래이크 포인트를 잡아 메모리나 다른 정보를 찾아서 해야 하는 것이 엔진 개발자의 일이다.

멘토님이 말하는 가상화 엔진은 디버거 등은 동적 분석 엔진이다.

한계가 너무 많다.

개발하기 너무 어렵다.

멘토님이 말하는 가상화 엔진은 수종이가 찾은 문서에서 힌트를 얻을 수 있음.

문서를 읽었을 때 메모리 변화가 어떨지 추정하는 것이다.

어플리케이션을 띄우기 위해 읽을 때 취약점을 찾을 수 있다.

최신의 악성 샘플을 잡을 수 있게 목표를 잡아도 성공한다.

어떻게 잡는지는 찾아봐야 한다.

멘토님은 파싱을 통해 찾아 낼 것 같다.

기존은 동적이므로 안되고 악성코드를 보고 원리를 파악해 찾는 것이다.

“리버스 핵심원리” 거의 필독서 이다.

기획 설계 코딩 테스트 배포

기획이나 설계없이 코딩을 많이 했다.

팀프로젝트 하면서 두서 없이 한적이 있나?

최종산출물이 제대로 나온지 없었다.

해당 단계를 지키는 편이다.

기획이나 설계까지 나올 것을 기대했다.

기획 설계가 반드시 필요하다.

꾸준히 잠재력과 노력을 쓸 수 있는 장치가 필요하다.

프로젝트에 대한 막연함을 다듬어주는 것이 기획이고, 이렇게 하면 될 것 같다는 신호를 받을 때가 기획의 마무리다.

이후는 기획과 코딩의 중간인 설계를 하는 것이다.  
코딩에 대한 클래스, 설계도를 어떻게 짤지 생각하는 것을 말함.

테스트와 배포는 제외.

기획: 가장 중요한 것은 단순화이다.

모든 도면은 화살표와 도면이다.

모든 다이어그램은 동그라미나 직선을 사용

Chart, bubble chart

Description automatically generated

이 둘을 나타내는 것이 백신과 서버이다.

내 피씨의 악성 문서가 동작하지 않게 탐지하는 것이다.

어떻게 진단은 빼고 진행.

Chart, bubble chart

Description automatically generated

이 둘 사이에는 백신은 검사요청, 진단 엔진은 검사 결과를 알려줘야 한다.

Chart, bubble chart

Description automatically generated

심플한 버전 다음에 어떤것을 생각해야 하는가?

이는 무엇을에 대한 답이다.

다음은 언제?

외부에서 다운로드 받은 문서가 처음 열릴 때 🡪 가장 이상적이다.

이를 어떻게 아나?, 다운로드 받았을 때 문서 검사는?

이를 수정한 것은 새로운 문서 이므로 검사할 필요 없다.

🡪 내용을 바탕으로 해쉬?  
🡺 사용자 관점에서 생각해본다. 나머지 생각 제외.

하나씩 요건을 제거하자. (언제)

진단 엔진에서도 생각하면 언제 검사하냐?

🡪 검사 요청이 왔을 때, 이전의 검사 이력이 없다면 새롭게 검사 이후 DB에 기록, 이전에 있다면 DB에서 검사 결과를 꺼내 반환

새로 업로드 된 샘플은 따로 보관

Chart, bubble chart

Description automatically generated  
Chart, bubble chart

Description automatically generated

여기에 백신에 무언가를 붙일 것이다.

따라서 추가로 필요한 것이 업데이터 이다.

업데이트는 새로운 백신 바이너리가 있을 때 백신을 종료하고 바이너리를 교체한후 실행

실행 중에 5분 간격으로 바이너리 확인

이에 대응하는 업데이트 서버가 있다.

Chart, diagram, bubble chart

Description automatically generated

이렇게 하나씩 늘리면서 필요한게 무엇인지 확인

백신에서 중요한 것이 데이터 베이스이다.

여기에 필요한 것이 블랙리스트 DB이다.

내부에서는 로컬 DB에 존재하지 않는 파일에 대해서만 분석을 요청해야 한다.

누군가 물어본거에 대해 물어보지 않게 계속해서 동기화 해야한다.

이런식으로 우리가 백신이란 것을 프로젝트에 대해 생김새에 대해 생각 했어야 했다.

Chart, diagram

Description automatically generated

이에 맞게 설계를 해야 한다.

고려해야하는 것은 가용자원은?, 각자의 역량과 역할은?

성민이는 분석 담당 🡪 알고리즘에 대한 참여를 한다고 열어주면 좋을 듯하다.

개발자 3명이 있다 가정한다면?

가장 top level설계는 다음과 같다.

Diagram

Description automatically generated

3개로 나눈 것은 3명이 맡아서 하자.

담당 포지션을 맞춰서 생각하고 기획으로 돌아가 가능 불가능을 판별한다.

이를 **컴포넌트 다이어그램**이라고 한다..

컴포넌트 래벨에서 어떻게 상호동작을 하는지?

화살표는 누가 누구를 아는지 이다.

즉 종속성을 이야기한다.

그 이후에 시퀀스 다이어그램이 나온다.

이는 보통 백신서버와 클라이언트는 소켓 통신을 해서 통신을 주고 받아야 한다.

그럴 때 클라이언트가 무언가 요청하고 서버가 응답하는 구조이다.

다양한 프로토콜이 있을 수 있음.

어떤 메시지냐?

Chart

Description automatically generated

프로토콜을 미리 구상하는 것.

이후는 정적 다이어그램이다.

이는 객체마다 달라진다. 즉 3가지를 만들면 된다. 해당 구조에서는

이후 여기서 해당 클래스 다이어그램을 그리면 된다.

코드로 작성할 클래스 이름과 메소드 이름을 작성.

OOP를 못하면 절차적으로 짠다면 순서도를 짜기.

문서의 목적은 구조를 잡는 것이다.

세세하게 할 필요 없다. 어떠한 메소드를 사용하는 지 쓰기.

문서를 고치는 것은 골격을 변경할 떄 사용.

Diagram

Description automatically generated

실제 사용 관련

정육면체가 하나의 pc이다.  
 Diagram

Description automatically generated with low confidence

Diagram

Description automatically generated

이렇게 만든 설계 문서가 발표자료에 사용된다.

반드시 거쳐야할 자료이다.

동적 다이어그램도 있다.

🡪 activity 다이어그램.

다음주에 기획문서와 설계문서에 대해 기획해오기.

절차적이던 OOP처럼 해야 한다.

기획은 찾은 논문에서 많은 힌트를 얻음.

기획을 시작하기 전에 나올 것이 요구사항 분석이다.

서비스는?

고객이 원하는 것은?

이에 대한 기획을 만들어야 한다.

남들과 똑같은 것은 만들 필요가 없다.

기획은 기술적인 생각을 버리고 하는 것이다.

처음의 기획은 최초의 요구사항으로부터 나온 것이다.

진단엔진이 사용할 수 있는 시간은?

남성엽 멘토님 관련 피드백

🡪 SW개발 관련해서 많이 안해봤어서 좋은 피드벡인 것 같다.

멘토님도 블록 다이어그램을 그리면서 했다.

스키메틱이나 설계도를 그리기전에 컴포넌트가 어떻게 되는지 이후 진행했다.

그런 것이 습관이 남아있어 취약점이나 퍼징 개발시 사용한다.

기술검토도 이번주에 마저 못한것도 검토해서 wbs도 나와야 한다.

Wbs는 제외하기.

기획 설계 마무리하고 기술적으로 검토해보기.