

# 심재빈 *Sim Jaebin*

스틸리언에서 5년째 iOS 보안 솔루션을 만듭니다. AI가 개발의 경계를 넘는 지금, 그 AI를 가장 잘 다루고 스마트한 일상까지 직접 만드는 개발자가 되려 합니다.

영문명	Sim Jaebin	생년월일	2001.04.24 (만 25세)
성별	남	이메일	simjabin@gmail.com
GitHub	github.com/jaebinsim	거주지	서울 동작구 노량진동
병역	산업기능요원 복무 완료 (2022.01 - 2023.12)		

<b>경력</b> <b>스틸리언</b> AppSuit iOS 보안 솔루션 제품군 설계·구현·운영 — 100+ 개 고객사 적용	<b>학력</b> <b>학점은행제</b> 컴퓨터공학 학사 · 2025.10~2026.10 (취득 예정)	<b>자격증</b> <b>총 10건</b> 정보처리기사 외	<b>포트폴리오</b> <b>총 51건</b> 경력 42 · 개인 9
---	---	--	--

## 기술 스택

언어	C Objective-C Python3 Shell Swift TypeScript
기술·라이브러리	Astro Binary Instrumentation Cloudflare Tunnel Cloudflare Workers Code Signing Dynamic Library Injection FastAPI Fat Binary Frida GitHub Actions GitLab CI/CD Jenkins Pipeline LC_SYMTAB LIEF LLM MVVM Mach-0 Notion API Ollama Puppeteer RAG RxSwift SSE Tailwind CSS XCFramework arm64e libimobiledevice ollama vLLM
환경·도구	Claude Code Hex Edit Hopper Jenkins Mach-0 Viewer PyCharm VSCode Xcode

## 자기소개서

### 보안 개발자가 되기까지

중1 때 겪은 DDoS 사고에서 시작해, 5년차 iOS 보안 솔루션 팀 리드가 되기까지

중학교 1학년 때 직접 운영하던 마인크래프트 서버가 DDoS 공격으로 마비된 적이 있습니다. 뉴스에서나 보던 일을 직접 겪고 나니 막연히 좋아하던 컴퓨터가 '지켜야 하는 것'으로 바뀌었고, 그때부터 보안을 업으로 삼겠다고 마음먹었습니다.

인문계 고등학교에 다니면서도 전공 활동을 스스로 찾아다녔고 특기자 전형으로 컴퓨터공학과에 입학했습니다. 다만 코로나로 온라인 수업이 이어지면서, 학교에 더 머무르기보다 빠르게 변하는 보안 시장에서 실무로 부딪히는 편이 낫다고 판단했습니다. 군 대체복무가 가능한 보안 기업 스틸리언에 입사해 iOS 보안 솔루션 개발을 시작했고, 학사는 학점은행제로 2026년 취득 예정입니다.

입사 초기에는 Objective-C 앱에만 적용되던 보호 기능을, Swift 문자열 암호화와 클래스 난독화를 직접 연구해 Swift 앱까지 확장했습니다. iOS가 기본 제공하지 않는 화면 캡처 방지 기능을 자체 연구해 특허 출원으로 이어가기도 했습니다. 이렇게 기능을 하나씩 늘려가다 macOS 보안 솔루션 신규 개발을 단독으로 맡았고, 이후 AppSuit 제품군의 iOS 파트 전반을 책임지게 됐습니다.

그렇게 5년이 지났습니다. 지금은 100여 개사에 적용된 iOS 보안 솔루션을 운영하면서 2025년 7월부터 iOS 팀을 이끌고 있습니다. 중학생 때 서버가 마비됐을 때 느낀 그 감각이, 매년 새로 등장하는 탈옥·후킹에 대응하는 지금의 동기와 크게 다르지 않습니다.

---

## 핵심 역량과 전문성

Mach-O 저수준 가공부터 멀티플랫폼·0→1 제품, 그리고 보안과 AI의 접점까지

제 전문성은 'iOS 보안 SDK를, 도입하는 개발자가 가장 적은 마찰로 적용하도록 만드는 일'에 있습니다.

가장 깊게 파고든 영역은 Mach-O 바이너리 저수준 가공입니다. 문자열 암호화나 API Hiding 같은 보호 기능은 본질적으로 바이너리에 임의 섹션이 박혀야 동작하는데, 이 작업을 빌드 후처리에 의존하던 구조에서 framework 빌드 시점 안으로 끌어왔습니다. 그 결과 고객사는 별도 빌드 인프라 없이 framework 하나만 추가하면 보안 SDK를 도입할 수 있게 됐습니다(AppSuit Air). Fat/Thin Binary, arm64e(Pointer Authentication), Xcode 26 Enhanced Security처럼 플랫폼·아키텍처가 바뀔 때마다 깨지는 지점을 흡수하는 설계도 직접 맡아 왔습니다.

폭도 넓혀 왔습니다. AppSuit Premium·Air 를 비롯해 캡처 방지·악성앱 탐지·보안 키패드 등 6개 제품의 iOS 파트를 0→1로 만들거나 장기 운영했고, 기능 개발을 넘어 시장·경쟁 제품 조사부터 가이드 문서까지 제품 한 사이클을 혼자 끌고 가 본 경험이 여러 번 있습니다.

최근에는 보안과 AI의 접점으로 영역을 넓히고 있습니다. 외부 LLM을 쓸 수 없는 보안 회사 특성에 맞춰 로컬 임베딩·리랭커 기반 사내 RAG 시스템과 GitLab MR 자동 리뷰봇을 직접 구축했고, 온디바이스 AI 모델(ONNX) 보호를 신규 PoC로 선점하고 있습니다. 필요한 도구는 사다 쓰기보다 직접 만들어 팀에 들였습니다.

---

## 일하는 방식과 협업

내가 없어도 굴러가는 운영 구조를 만드는 플레이어형 팀장

저는 '제가 없어도 굴러가는 구조'를 만드는 일을 중요하게 생각합니다. 2025년 7월 iOS 팀 팀장을 맡은 뒤 가장 먼저 한 일도, 머릿속에 있던 운영 방식을 팀의 구조로 옮기는 것이었습니다.

제품별 메인·보조 담당을 명시한 배정 체계를 만들고, 운영과 신규 개발의 비중을 7:3으로 고정해 고객 이슈 대응에 매몰되지 않으면서도 새로운 제품을 향한 동력이 끊기지 않게 했습니다. 팀의 목표·업무 분장·제품 보드·할 일 트랙을 Notion과 Jira에 표준화해, 누구든 그 구조만 보면 지금 무엇이 어떤 일정으로 돌아가는지 알 수 있게 했습니다.

협업에서 제가 가장 신경 쓰는 부분은 '검증 가능한 절차'입니다. 149개사에 적용된 솔루션을 운영하다 보면 입장이 제각각인 이슈가 계속 들어오는데, 재현을 공유 가능한 샘플로 좁히고 원인·패치·회귀 검증을 단계별 산출물로 고정해 두면 담당자가 바뀌어도 같은 품질로 대응할 수 있습니다. QA 역시 사람이 매번 챙기는 대신 빌드-설치-실행-크래시 검증을 자동화해, 팀이 6개월 주기 메이저 릴리즈를 안정적으로 지킬 수 있게 했습니다.

다만 저는 관리만 하는 팀장이 되고 싶지는 않습니다. 팀장을 맡은 뒤에도 설계와 구현을 직접 병행하는 플레이어형 팀장으로 일하고 있고, 그것이 팀원과 같은 언어로 대화하며 신뢰를 쌓는 방법이라고 생각합니다.

---

## 앞으로 되고 싶은 개발자

보안에서 단련한 SDK·플랫폼 설계 감각으로, 경계를 넘는 SDK를 만드는 개발자

제가 5년간 단련한 건 보안 지식 자체라기보다, 플랫폼과 아키텍처의 차이를 흡수해 도입하는 쪽의 고민을 줄여 주는 설계 감각이었습니다. 마침 그 무대가 보안이었을 뿐입니다.

그래서 앞으로도 보안 직군에만 머물기보다 SDK·플랫폼 엔지니어로 폭을 넓히고 싶습니다. 한 줄 추가만으로 동작하는 SDK가 되기까지, 내부에서는 Fat/Thin 바이너리·arm64e·OS와 컴파일러 버전 차이처럼 깨지기 쉬운 지점을 얼마나 흡수해야 하는지를 여러 제품에서 부딪혀 봤습니다. iOS와 macOS를 한 코드베이스에서 양립시키고 빌드 통합 자체를 없앤 경험은, 결국 경계를 넘는 SDK를 어떻게 안정적으로 입힐 것인가의 문제였습니다.

플랫폼 경계를 흡수하는 SDK를 만드는 일이라면, 도메인이 무엇이든 제가 해 온 작업과 그대로 맞닿아 있다고 생각합니다. 보안에서 익힌 저수준 이해와 SDK 설계 감각을 새 무대에서 이어가고 싶습니다.

덧붙여, 만든 것을 글과 코드로 남기는 일도 함께 해 왔습니다. 사내에는 외부 LLM을 못 쓰는 환경에 맞춘 RAG 시스템과 코드 리뷰 봇을 직접 들였고, 밖으로는 멀티 에이전트 OSS를 만들고, 비개발자용 개발 도구 활용서를 썼습니다. 결국 만들고, 그걸 남기는 개발자이고 싶습니다.

2021.06 - 재직중 · 5년 1개월 서울

## 스틸리언

iOS Team Lead

AppSuit iOS 보안 솔루션 제품군 설계·구현·운영 — 100+ 개 고객사 적용

## 학력

---

재학중

### 학점은행제

컴퓨터공학 학사 · 2025.10~2026.10 (취득 예정)

---

종퇴

### 고려대학교 (세종)

컴퓨터융합소프트웨어 · 2020.03~2025.08 (특기자 전형)

---

졸업

### 퇴계원고등학교

이과계열 · 2017.03~2020.02

## 자격·수상

---

### [자격증]

2025.06

#### 정보처리기사

한국산업인력공단

---

2024.06

#### 정보처리산업기사

한국산업인력공단

---

2021.12

#### 정보처리기능사

한국산업인력공단

---

2021.03

## 1종보통 운전면허

경찰청 (도로교통공단)

---

2019.04

### 네트워크관리사 2급

한국정보통신자격협회

---

2019.02

### 3D 프린터조립전문가 2급

3D프린팅산업협회

---

2018.06

### 리눅스마스터 2급

한국정보통신진흥협회

---

2016.07

### 정보기술자격 (ITQ) 한글엑셀 (한셀) A등급

한국생산성본부

---

2011.12

### 정보기술자격 (ITQ) 한글파워포인트 (한쇼) B등급

한국생산성본부

---

2011.10

### 정보기술자격 (ITQ) 인터넷 A등급

한국생산성본부

## [수상]

2024.08

### let us:Go! 2024 해커톤 대상 — Vision Desk (Apple Vision Pro AR) · let us:Go! 운영위원회

let us:Go! 2024 해커톤 대상(4인 팀) — Apple Vision Pro AR 작업대 'Vision Desk' 개발

let us:Go! 2024 해커톤 (4인 팀) 대상 수상 / 작품명 Vision Desk — Apple Vision Pro 를 통해 가상 작업대를 생성하여 캘린더·리마인더·사진·지도 정보를 AR 기술로 확인하는 앱 / 경진 URL: letusgo2024-summer.vercel.app / 소스 github.com/letusGo-Hack/Team\_6\_HomP\_XR\_Min

---

2020.04

### 고려대 세종 KUDDING 프로그래밍 경진대회 수상 (1회 · 3회, 2020) · 고려대학교 세종캠퍼스

고려대학교 세종캠퍼스 교내 KUDDING 프로그래밍 경진대회에서 2회 수상 (제 1 회 2020.10, 제 3 회 2020.12)

1학년 시절 (2020) 고려대 세종캠퍼스 교내 KUDDING 프로그래밍 경진대회 / 제 1 회 (2020.10) · 제 3 회 (2020.12) 2회 수상

---

2019.01

송실대 고교-대학연계 심화과정 알고리즘 분석·코딩 최우수상 (2019) · 송실대학교

송실대 고교-대학연계 심화과정 (컴퓨터과학 1) 알고리즘 분석·코딩 평가 최우수상 (1위)

고교 재학 중 송실대 고교-대학연계 심화과정 (UP) 컴퓨터과학 1 수강 / 알고리즘 분석·코딩 평가 최우수상 (1위)

## 활동·집필·미디어

---

2026.06

『비개발자를 위한 클로드 활용 가이드』 단독 저자 · 코리아교육그룹

Claude 를 코딩 없이 업무에 녹여 쓰는 패턴을 비개발자 독자 대상으로 정리한 단독 저서. 2026.06 출간 예정.

단독 집필 · 출판 계약 완료 (2026.06 출간 예정)

---

2026.01

유튜브 자취남 채널 출연 · 유튜브 자취남

직접 구축한 스마트홈·IoT 자취 환경을 소개한 '자취남' 채널 출연 — 게시 후 37만뷰

유튜브 '자취남' 채널 'IOT 천재의 집' 편 출연 (2026.01.30 게시) · 37만뷰

---

2025.11

인도네시아 고객 교육 출장 · 스틸리언 / 인도네시아 현지 고객

인도네시아 현지 고객 교육 출장(2025.11) — iOS Premium 제품 현장 교육 및 고객 취약점 대응 병행

1인 해외 출장 교육 — iOS 보안 솔루션(Premium 라인) 현장 교육 / 취약점 대응 병행

---

2025.03 – 2025.11

KISIA S-개발자 3기 대표 멘토 · 한국정보보호산업협회 (KISIA)

KISIA S-개발자 3기 대표 멘토 (2025.03~2025.11) — 멘티 팀 프로젝트 Swingft (LLM + Swift AST 기반 코드 변환) 지도

KISIA 공식 프로그램 / 9개월 수행 / 멘티 팀 프로젝트 Swingft 지도

---

2023.06 – 2023.12

특허 출원 — 텍스트 필드 보호 속성을 이용한 스크린 캡처 및 미러링 방지 장치 및 방법 · (주)스틸리언

iOS UIView / Layer 계층 조작 + 텍스트 필드 보호 속성 확장으로 스크린 캡처·미러링을 차단하는 기술

특허 공동 출원 1건 (출원인 □스틸리언, 공동발명자)

---

2021.03 – 2021.06

고려대 세종 제4회 KUDING 프로그래밍 경진대회 운영진 · 고려대학교 세종캠퍼스 컴퓨터융합소프트웨어학과

고려대 세종 컴퓨터융합소프트웨어학과 주관 제4회 KUDING 프로그래밍 경진대회 운영진 (2021.03~2021.06) · 문자열·정렬 알고리즘 문제 설계·출제 및 채점·현장 운영 담당

고려대 세종 컴퓨터융합소프트웨어학과 주관 프로그래밍 경진대회 운영진 참여 / 대회 문제 출제·채점·현장 운영·질의응답 담당 / 문자열 처리 (String)·정렬 (Sort) 알고리즘 문제 직접 설계·출제

---

2019.03 – 2019.07

아두이노 지각자 체크기 제작 (퇴계원고 수업 프로젝트, 2019) · 퇴계원고등학교

학급 지각자 체크용 아두이노 기기 제작 (2019.03—07, 고등학교, 9인 팀)

Arduino + C / 9인 팀 / 지하철 게이트를 모티브로 하여 아두이노를 활용한 학급 지각자 체크기 개발

---

---

2019.01 - 2019.02

**송실대학교 고교-대학연계 심화과정 (컴퓨터과학 1) · 송실대학교**

고교 재학 중 송실대학교 고교-대학연계 심화과정 (UP) 컴퓨터과학 1 수강 — Python 프로그래밍 알고리즘 기초·분석 수료

고교 재학 중 대학연계 심화과정 수강 / Python 알고리즘 기초·분석 / 수강 결과로 알고리즘 분석·코딩 최우수상 (별도 카드)

---

2018.03 - 2020.02

**퇴계원고 프로그래밍 동아리 부장 · 퇴계원고등학교**

퇴계원고등학교 프로그래밍 동아리 부장 (2018.03~2020.02) — 부원 대상 C 언어·아두이노 수업 + 굴절·반사율 계측기 제작 주도

동아리 부원 대상 C 언어 기초 프로그래밍 및 아두이노 회로 설계 수업 진행 / 아두이노를 활용한 굴절·반사율 계측기 설계 및 제작 프로젝트 주도 / 학생동아리 융합과학 상설부스 대회에 학교 대표로 출전

# 경력 기술서

## [메인 프로젝트]

2024.06 - 2024.12 · 7개월

### AppSuit Air — IPA 업로드만으로 적용하는 SaaS 보안 솔루션

AppSuit Premium 보안 기능을, 호스트 앱 빌드 통합 없이 IPA 업로드만으로 적용하는 SaaS 보안 솔루션 — iOS 1인 단독 개발

역할 협업 프로젝트의 iOS 파트를 1인 단독 담당 — 기획·설계·코어 개발·서버 연동·테스트·패키징 전 과정 단독 수행 및 후속 담당자 인수인계 주도

임팩트 2024.06~12 iOS 1인 단독 개발, 7개월 내 제품화(2024.12 출시, 2025.12 인수) · 정적 라이브러리를 직접 넣고 빌드 설정을 손보는 수작업 없이 IPA 업로드만으로 보안 SDK 도입 가능 → Premium 적용 영역을 도입 고객 전반으로 확장 · 빌드 후처리에 의존하던 보호 기능(문자열 암호화·API 호출 은닉·리소스 복호화)을 framework 내부에 내재화 · 카카오톡·유튜브 등 일반 상용 앱 IPA 적용 검증

기술

Swift Objective-C Python3 Mach-O LIEF Binary Instrumentation Dynamic Library Injection  
Code Signing Fat Binary

## [코어 운영]

2026.05 - 2026.08 · 4개월

### Premium iOS 대규모 리팩토링 (Objective-C → Swift / 모듈화 / AI 개발 워크플로우 엔지니어링)

Premium iOS 레거시 Objective-C 코드베이스를 Swift·모듈화 구조로 전환하고 AI 개발 워크플로우(TextRAG·MR 리뷰봇)를 적용하는 장기 리팩토링 트랙 설계 주도

역할 Premium iOS의 Objective-C → Swift 전환·모듈 경계 재설계·AI 개발 워크플로우 연동(TextRAG corpus 최신화, GitLab MR 리뷰봇 활용)을 포괄하는 대규모 리팩토링 트랙의 계획·설계 주도. iOS 팀 공동 적용 전제.

임팩트 2026년 5월 정기 릴리즈 출시(2026-05-15) 직후 착수하는 본인 주도 장기 트랙으로 설계. Objective-C → Swift 전환과 모듈화를 병행 설계, TextRAG·GitLab MR 리뷰봇을 리팩토링 일부 자동화 자원으로 재사용. 수작업 검토·리뷰 공수 절감을 기대 목표로 설정(트랙 착수 전, 미검증).

기술

Swift Objective-C XCFramework

2026.03 - 2026.05 · 3개월

### Premium iOS v26.1.0 사용성 개편 (2차 옵션 UX·스레드 컨벤션·빌드 페이지)

Premium iOS 2026년 5월 릴리즈에서 2차 옵션 UX(정책별 동적 노출), 스레드 네이밍 공통 컨벤션, 빌드 페이지 UX 세 영역을 개편

역할 Premium iOS 사용성 3대 개편(2차 옵션 UX 개선, 스레드 네이밍 공통 컨벤션 수립, 빌드 페이지 UX 개편 안내 작성)을 설계부터 구현, 내부 공유 문서 작성까지 수행. 구현 방향은 1:2차 팀 회의로 합의.

임팩트 2차 옵션 사용성 개선 담당(2026-03-10~05-19 해결) / 탐지 정책 용어 4종 표기 통일 / 스레드 네이밍 공통 컨벤션 수립(2026-04-30~05-01)으로 크래시·디버그 시 기능 추적 1시간 이상에서 수 분으로 단축 / 빌드 페이지 UX 개편 안내 문서 작성(2026-05-04~05-07) / 개편 결과는 Premium 도입 고객 전반에 적용

기술

Swift Objective-C Python3 XCFramework MVVM

2025.12 - 2026.04 · 5개월

### Cloud → Premium 명칭 통일 작업 (제품명·가이드)

AppSuit 제품 라인의 'Cloud → Premium' 명칭 통일을 2026년 4월 메이저 릴리즈 일정에 맞춰 일괄 적용한 iOS 팀 협업 트랙 (Premium iOS 가이드 본인 단독 완료)

역할 AppSuit 제품 라인 명칭 통일(Cloud → Premium) 팀 협업 참여. Premium iOS 가이드 개편 단독 완료, 2026년 4월 메이저 릴리즈와 교차 일정으로 명칭 변경 일괄 적용 (2026-04-10).

임팩트 Cloud → Premium 명칭 통일 작업 (2026-04-10), 메이저 릴리즈와 교차 일정 / Premium iOS 가이드 개편 본인 단독 완료 / 선행 트랙으로 내부 프로젝트 명칭 적용 2025-12-16~2026-01-16 완료

2025.11 – 2026.01 · 3개월

### Premium v26.0.0 릴리즈 사이클 (2026.01 출시)

AppSuit Premium iOS 2026년 1월 메이저 릴리즈 사이클 수행 및 후속 릴리즈 설계·QA 체계 개선 참여

역할 iOS 팀 일원으로 메이저 릴리즈 사이클의 컴파일러 최적화 검증(팀 공동), QA, 팀 릴리즈 조율에 참여. 후속 QA Board 개선과 예외처리 룰 가이드 정비는 본인 단독 수행.

임팩트 2026.01.22 메이저 릴리즈 출시 달성(iOS 팀 협업) / 컴파일러 최적화 검증(2026.01.22~28, 팀 공동) / Premium QA Board 개선 (2026.02.02~06, 단독) / 예외처리 룰 가이드 정비(2026.02.02~10, 단독) / CLL\_XcodePlugin 호환 업데이트(2026.05.14) 후 2026.05.15 후속 릴리즈 출시

기술 Swift Objective-C Python3 Mach-O XCFramework

2025.08 – 진행 중 · 11개월

### arm64e / Xcode 26 Enhanced Security 지원 연구

Xcode 26 강화 보안 옵션이 강제하는 통합 바이너리(arm64+arm64e) 환경에서도 AppSuit Premium 6개 모듈이 동작하도록 빌드 후처리 인프라를 선행 확장한 연구·구현

역할 Xcode 26 강화 보안(Enhanced Security)·통합 바이너리(Fat Binary) 대응 선행 연구 — 대응 시나리오 비교표(의사결정 매트릭스) 작성, 통합 바이너리 후처리 객체 설계·적용, Premium 6개 모듈 호환 검증 가이드 작성. iOS 팀 정기 회의에서 시나리오 검토 공유.

임팩트 4개 대응 시나리오 비교표 작성으로 고객사 요청 시 1~2주 내 대응 가능 환경 마련 / 후처리 도구에 통합 바이너리(Fat Binary) 처리 객체 신설(단일 슬라이스·통합 바이너리 양쪽에 동일 패치 단계 적용) / Premium 6개 모듈 arm64e 호환 비교표 및 라이브러리 빌드 가이드 정리

기술 Swift Objective-C Python3 Mach-O LIEF arm64e Fat Binary Binary Instrumentation

2025.08 – 2025.10 · 3개월

### 신규 OS 대응 (iOS 26 / Xcode 26 / Swift 6 / os\_log 전환)

Apple iOS 26 / Xcode 26 / Swift 6.2 정식 출시에 맞춰 Premium·서브 제품군 6개를 일정 안에 일괄 대응하고 제품별 호환 매트릭스 정리

역할 iOS 팀 공동 대응에서 팀 명세 공유·항목 조율·릴리즈 일정 조율 주도. 가이드 Enhanced Security 미지원 명시, NSLog → os\_log 전환, 제품별 Swift 6 호환 매트릭스 작성 담당.

임팩트 Premium 정규 패치 릴리즈 (8월 말 ~ 9월 초 목표 달성) / Anti-Capture·VPNBlock·AVSuit·Keypad 등 6개 제품군 일괄 업데이트 조율 / 가이드 Enhanced Security 미지원 명시 추가 / 샘플·스니펫 NSLog → os\_log 일괄 전환 / 제품별 Swift 6 호환 매트릭스 작성

기술 Swift Objective-C XCFramework arm64e

2025.07 – 2025.12 · 6개월

### Premium iOS v25.5.0 SSL Pinning 기능 설계 (4모듈 파이프라인)

iOS 내장 SSL Pinning(NSPinnedDomains·URLSessionDelegate)을 래핑해 고객이 호스트·인증서만 등록하면 MITM을 차단하는 기능을 4모듈 파이프라인으로 설계·출시

역할 Premium iOS SSL Pinning 기능을 정적 라이브러리 런타임 검증 + 빌드 후처리 + macOS 데스크톱 도구 + 재서명 자동화 플러그인 4모듈 공동 파이프라인으로 설계 주도. iOS 내장 SSL Pinning 래핑 방식 의사결정 담당.

임팩트 Premium iOS 2025년 9월 릴리즈(2025-09-26 출시)에 SSL Pinning 기능 정식 포함 / 4모듈 파이프라인 설계 주도 / 공개키 기반 방식 채택으로 해외 앱 심사 체크리스트 대응 / Charles Proxy 기반 MITM 검증 절차 5단계 표준화 / 2025-12-08 인증서 등록·재등록 흐름 안정화 개선

기술 Swift Objective-C Python3 XCFramework Code Signing Binary Instrumentation

2025.01 – 진행 중 · 1년 6개월

### Premium 기능 명세·고객 가이드 문서화 (2025~)

Premium 제품군 7종 이상의 기능 명세·고객 가이드를 재사용 가능한 안내 템플릿으로 구조화·문서화 주도

역할 Premium 제품군 기능 명세·고객 가이드 문서화 주도 — 제품별 가이드·릴리즈 노트·고객 안내 템플릿 구조 표준화, 세부 작성은 팀 협업

임팩트 Premium 제품군 7종 이상 가이드 구조 표준화 / AppSuit Premium iOS 가이드 릴리즈 주기에 맞춰 지속 업데이트 / AppSuit Air 가이드 신규 작성(2026.03, 팀 협업) / iOS 팀 제품 안내 종합본 33장 본인 주도 정리(제품 개요·기능별 설명·고객 검증 절차·SSL Pinning·탐지 정책 포함)

2025.01 - 2025.12 · 1년

## AppSuit Keypad iOS 제품군 재정비 (Legacy + 해외 금융 고객사 + Crypto 내장화)

레거시 AppSuit Keypad 제품군(해외 금융 고객사 적용)을 '다음 담당자가 이슈 단독 대응 가능' 수준으로 재정비 주도

역할 AppSuit Keypad 제품군(Legacy Core·대만·인도네시아 금융 고객사 라인) 재정비 단독 주도. 고객 이슈 대응·유지보수를 위해 Core 프로젝트 호환화 및 라이브러리 패키징 트랙 정리(Crypto 내장화·Swift 6 리뉴얼은 팀원 메인 병행).

임팩트 iOS 팀 2025 목표의 본인 항목 'AppSuit Keypad(Legacy·해외 금융 고객사) 제품군 재정비' 명시 / Legacy Core 재정비(대만·인도네시아 라인 호환화)·대만 금융 고객사 라인 재정비 본인 담당 / 대만 금융 고객사 긴급 이슈 패치본 전달(2025-10) / 인도네시아 금융 고객사 이슈 교차 대응 / 이관 기준 '다음 담당자가 고객 이슈 단독 해결 가능' 충족

기술 Swift Objective-C XCFramework

2023.07 - 2023.09 · 3개월

## iOS API Hiding 연구 (2023-07~09, 리팩토링·성능 최적화)

AppSuit Premium iOS의 API Hiding 기능을 새 Xcode 빌드 환경·바뀐 Mach-O 구조에 맞춰 재작성하고 속도를 끌어올린 연구

역할 AppSuit Premium iOS의 API Hiding 기능을 새 Xcode·Mach-O 구조에 맞춰 재작성하고 속도를 최적화. ARM64 어셈블리에서 API 호출 분기를 바꿔, 원래 호출 대상과 코드 흐름(제어 흐름 그래프, CFG)을 분석 도구가 읽지 못하게 막도록 설계.

임팩트 약 3개월 1인 수행 / ARM64 어셈블리의 API 호출 분기를 바꿔 원래 호출 대상·코드 흐름(CFG) 식별 차단 / Mach-O 바이너리 안의 어셈블리를 읽고 고치는 Python 도구 직접 개발 / 이후 'Dynamic API Hiding' 정식 제품 기능으로 자리 잡음

기술 Objective-C Python3 Mach-0 Binary Instrumentation Dynamic Library Injection

2023.03 - 2023.05 · 3개월

## Swift 문자열 암호화 + 클래스 난독화 + Dynamic API Hiding (2023 완성)

Swift 앱의 문자열·클래스 이름·시스템 API 호출이 분석 도구에 그대로 드러나던 공백을, 문자열 암호화·클래스 이름 난독화·동적 API 숨김 3종 보호 기능으로 처음부터 구현해 AppSuit Premium에 정착

역할 문자열 암호화 / 클래스 이름 난독화(분석 도구에서 의미 없는 이름으로 보이게) / 동적 API 숨김(호출하는 시스템 기능 이름을 실행 중에만 해석) 3종 기능을, 보안 코어 모듈(정적 라이브러리)과 빌드 후처리 단계 2모듈 조합으로 처음부터 설계·구현. iOS 바이너리(Mach-O ARM64)를 직접 뜯어보는 분석 도구를 자체 제작해 검증·QA 자동화·후속 인력 지도에 활용.

임팩트 3종 보호 기능(문자열 암호화 / 클래스 이름 난독화 / 동적 API 숨김) 신규 구현 / iOS 팀 제품 명세서·고객 가이드에 정식 포함 / 2026.01 iOS 팀 회의에 검증 절차 공식 수록 / AppSuit Premium 도입 고객 전반 적용 가능

기술 Swift Objective-C Python3 Mach-0 LIEF Binary Instrumentation LC\_SYMTAB

2023.01 - 진행 중 · 3년 6개월

## Premium Framework 보안 기능 연구 (Symbol Delete / Log Delete / Exclude Rule / 문자열 암호화 검증)

AppSuit Premium Framework의 보안 기능 4종(Symbol Delete·Log Delete·Class 난독화 Exclude Rule·문자열 암호화 검증)을 본인 주도로 설계·연구

역할 AppSuit Premium 보안 기능 연구를 다수 주도 — Symbol Delete(릴리즈 빌드 심볼 제거), Log Delete(디버그 로그 자동 제거), Class Name Obfuscation Exclude Rule(클래스 난독화 제외 규칙), 문자열 암호화 검증 파이프라인의 설계·연구를 단독 주도하고 구현은 팀과 협업.

임팩트 2023년 구현 이후 현행 릴리즈까지 지속 운영·개선. 4종 기능(릴리즈 빌드 심볼 제거로 정적 분석 난도 상승, 디버그 로그 자동 제거, 클래스 난독화 제외 규칙(Exclude Rule), 문자열 암호화 검증) 설계·연구 주도

기술 Swift Objective-C Python3 Mach-0 LIEF Binary Instrumentation LC\_SYMTAB

2021.10 - 진행 중 · 4년 9개월

## AppSuit Premium iOS 장기 운영·유지보수 (2021-10~현재)

AppSuit Premium iOS·앱 재서명 도구·CI/CD 빌드 파이프라인 플러그인의 장기 운영·고객사 이슈 트러블슈팅을 4년 이상 주도

역할 AppSuit Premium iOS 제품군의 장기 운영·고객사 이슈 트러블슈팅 주도 — 보안 코어 모듈(정적 라이브러리)·빌드 후처리 단계·앱 재서명 도구·Xcode 빌드 파이프라인 플러그인 라인 운영 전반 담당.

임팩트 AppSuit Premium iOS 도입 고객 전반을 여러 메이저 릴리즈에 걸쳐 4년 이상 연속 유지보수 / 앱 재서명 도구·자동화 빌드 파이프라인 플러그인 유지보수 / 고객사 이슈 대응 표준 프로세스(접수→재현→원인 분석→패치→회귀 검증→회신) 수립 / 고객사별 커스텀 패치 관리 표준(노션 기반 문서화) 정립

기술 Objective-C Python3 Mach-0

2021.06 – 진행 중 · 5년 1개월

### 취약점 대응 연구 (Dopamine 2 / Bootstrap / RootHide / Vnodebypass / TrollStore / Frida)

iOS 탈옥·훅킹 신규 변종 탐지 로직을 Premium에 반영하고 해외 고객 취약점 회신을 조율하는 운영 트랙

역할 iOS 탈옥·훅킹 도구 신규 변종(Dopamine 2 / Bootstrap / RootHide / Vnodebypass / TrollStore / Frida 포트 변경) 대응 트랙의 의사결정·고객 회신 조율 주도. 상세 연구는 팀원 분담, 본인은 Premium 릴리즈 일정과의 우선순위 조율·일부 탐지 정책 설계 참여.

임팩트 Premium 2025년 9월 릴리즈에 Dopamine 2·RootHide 탈옥 탐지 기능 정식 추가(2025-09-26) / Vnodebypass 대응 연구 2건 탐지 적용 후 종결 / 해외 고객 취약점 회신 조율(2025.07) / Frida 포트 변경 우회 대응·Bootstrap 탐지 연구 등록

기술 Swift Objective-C Python3 Mach-0 Binary Instrumentation Frida

## [신규 제품]

2026.03 – 2026.05 · 3개월

### AiSuit / AppSuit for AI Model — 온디바이스 AI 모델 보호 (ONNX PoC 통합)

iOS 앱에 탑재되는 외부 AI 모델 파일을 고객사 Xcode 수정 없이 AES-256-GCM으로 암호화해 보호하는 신규 제품 PoC를 1인 단독 수행

역할 AppSuit Air가 쓰던 '앱 빌드가 끝난 IPA 파일에 후처리만 더하는' 방식을 AI 모델 보호 영역으로 확장한 신규 제품 개념 검증(PoC) — 기획·설계·구현·검증·제안서까지 전 과정 담당, iOS 프레임워크 + Python 명령줄 도구(CLI) + 웹 콘솔(FastAPI) + 제안서 PDF 2종 산출

임팩트 PoC 완성. CLIP ViT-B/32 ONNX(336MB) 대상 실기기 검증에서 보호 적용 후에도 추론 결과 완전 동일(top-1 정확도 99.4%), 추론 지연 사실상 무손실. 8개 포맷 지원(ONNX·CoreML·TFLite·GGUF·PyTorch 계열). 제안서 PDF 2종(내부 21p·외부 12p)·FastAPI 웹 콘솔 산출, macOS·Linux 양쪽 서버 동작

기술 Swift Objective-C Python3 Mach-0 LIEF Dynamic Library Injection Code Signing XCFramework FastAPI

2024.05 – 진행 중 · 2년 2개월

### AppSuit VPNBlock iOS — 신규 제품 (VPN/Proxy 탐지)

iOS VPN/Proxy 활성 여부 탐지 Framework 를 시장 조사부터 기능 고도화·가이드·오픈 페이지까지 신규 구축

역할 iOS VPNBlock 신규 제품을 설계·구현 — 타 솔루션 조사, VPN 앱별 탐지 매트릭스 작성, 기능 고도화, 가이드·오픈 페이지 작성까지 전 일정 단독 수행.

임팩트 신규 구현(2024.05.02~06.25 완료) / 타 솔루션 조사 작성 / VPN 앱별 탐지 매트릭스 표 작성 / Proxy 탐지 추가 등 기능 고도화 / 최근 릴리즈에서 Swift 6.2 컴파일 고도화 후 안정 운영

기술 Swift Objective-C XCFramework

2024.05 – 2024.09 · 5개월

### AVSuit iOS — 신규 경량 탐지 SDK (탈옥·디버깅·Frida·훅킹)

AppSuit Premium 도입이 부담스러운 iOS 고객사를 위해, 탈옥·디버깅·Frida·훅킹 4종을 잡아내는 가벼운 탐지 SDK(AVSuit)를 처음부터 설계·구현

역할 Premium(풀 기능 보안 제품) 도입이 부담스러운 iOS 고객층을 겨냥한 경량 탐지 SDK(AVSuit)를 단독 개발. 탈옥·디버깅·Frida(앱 내부 동작을 가로채 조작하는 분석 도구)·훅킹(함수를 가로채는 변조 기법) 4종을 잡아내는 정적 라이브러리를 설계·구현. 탐지 결과를 매번 다른 값으로 돌려주는 방식을 설계해, 공격자가 탐지 함수를 가로채 무력화하는 우회를 차단. 고객 적용 가이드와 안내 페이지 작성까지 전담.

임팩트 2024.08~2024.09 개발 완료, 2024.09 출시. 탈옥·디버깅·Frida·훅킹 4종 탐지 정적 라이브러리 구현. 별도 후처리 도구 없이 헤더 파일과 라이브러리만 묶어 배포해, 고객은 빌드에 추가만 하면 적용. 2026.02 패치 이후 팀원 담당으로 이관

기술 Swift Objective-C XCFramework MVVM Frida

2023.11 - 2024.01 · 3개월

## AppSuit Premium macOS — 제품 라인 확장 및 AppSuitSign 리사이너

AppSuit Premium iOS 라인의 macOS 확장 — 정적 라이브러리 + 후처리 도구 2단 구조 이식 및 .pkg 재서명·리사이너 자동화 추가

**역할** AppSuit Premium의 iOS → macOS 제품 라인 확장을 주도하여 정적 라이브러리 + 후처리 도구 2단 구조 이식 설계·구현. 이 과정에서 AppSuitSign에 .pkg 재서명 자동화를 추가하고(2023.11) 앱스토어 배포용 .pkg 리사이너를 단독 구현.

**임팩트** Premium macOS 제품 라인(정적 라이브러리 + 후처리 + Utilities) 구조화 / macOS Big Sur ~ Sonoma 4개 메이저 버전 지원 범위 확보 / AppSuitSign 2023-11-13 릴리즈에 macOS .pkg 재서명 자동화 추가 + 앱스토어 배포용 .pkg 리사이너 단독 구현(2024-01-03 완료)

**기술** Python3 Shell Objective-C Mach-0 Code Signing

2021.06 - 진행 중 · 5년 1개월

## AppSuit Anti-Capture iOS — 연구·제품화·특허 출원

iOS View·Layer 계층 조작으로 앱 화면 스크린샷·녹화·미러링을 차단하는 SDK — iPhone Mirroring 탐지 연구와 텍스트 필드 보호 속성 특허 출원 담당

**역할** AppSuit Anti-Capture iOS 공동 운영 담당 — iPhone Mirroring(iOS 18) 탐지 1인 단독 연구 주도, 원격 제어 툴 대응 검증 주도, 텍스트 필드 보호 속성 기반 확장 메커니즘 특허 출원자.

**임팩트** 특허 출원 1건('텍스트 필드 보호 속성을 이용한 스크린 캡처 및 미러링 방지 장치 및 방법') / iPhone Mirroring(iOS 18) 탐지 1인 1주 단독 연구(2024.07.12~07.19) — 공식 API 부재 환경에서 3개 대체 경로 비교·권장안 도출 / 원격 제어 툴 대응 검증 및 고객 안내 표준 템플릿 정립(2023.06)

**기술** Swift Objective-C XCFramework RxSwift MVVM

## [DevOps·도구]

2025.01 - 진행 중 · 1년 6개월

## AppSuit Toolbox 기획 (2025)

고객사 현장 엔지니어(SE)가 AppSuit 적용 전 단계에서 쓰는 macOS 도구들을 단일 패키지로 묶는 AppSuit Toolbox 기획·설계 주도

**역할** 고객사 현장 엔지니어(SE)가 AppSuit 적용 전 단계에서 사용하는 macOS 도구들을 하나의 통합 패키지로 묶는 AppSuit Toolbox의 기획·설계 주도. 제품 접근성 관점에서 분산된 고객 워크플로우를 단일 도구로 정렬하는 구조 설계.

**임팩트** 고객 SE의 분산된 앞단 작업을 단일 macOS 패키지로 통합 / 고객이 한 곳에서 워크플로우를 적용하는 '단일 진입점' 구조 제공 / 1차 기능 완료 후 고객사 요구 채널 기반 확장 지속

**기술** Swift Python3 Code Signing

2023.11 - 진행 중 · 2년 8개월

## macOS App Package Tool (.app → .pkg 패키징)

macOS 앱 .app → .pkg 패키징·서명 자동화 도구 설계·운영

**역할** macOS 앱의 .app → .pkg 패키징·서명 자동화 도구 설계·운영 주도. Python3·Shell 기반 파이프라인 구성 및 AppSuitSign 연계 워크플로우 제공.

**임팩트** macOS .app → .pkg 패키징·서명 자동화 / AppSuitSign 연계 고객 SE용 워크플로우 제공 / 2023.11~ 운영 중

**기술** Python3 Shell Code Signing

2023.06 - 진행 중 · 3년 1개월

## Jenkins 기반 자동화 CI/CD

iOS 제품군 빌드·배포·자동 QA의 Jenkins 파이프라인 설계·운영 주도

**역할** Jenkins 기반 iOS 제품군 빌드·배포·자동 QA 파이프라인 설계·운영 주도. iOS QA Board 연동으로 릴리즈·고객 이슈 대응까지 자동화 범위 확장.

**임팩트** iOS 제품군 7종 이상의 빌드·탐지 검증·IPA 설치·로그 수집 자동화 / TestFlight·내부 배포 연동 / iOS 팀 QA Board와 연동한 릴리즈·고객 이슈 대응 자동화

2023.05 - 진행 중 · 3년 2개월

### GitLab CI/CD 구축

iOS 팀 GitLab CI/CD(코드 변경마다 자동으로 빌드·검사하는 체계) 구축·운영 주도 — Git Flow 연동, 병합 요청 시 자동 빌드·코드 검사, 리뷰봇 연계

역할 iOS 팀 GitLab CI/CD 구축·운영 주도 — 개발 프로세스 정비를 기술로 이행. 제품 저장소 전반의 파이프라인(자동 빌드·검사 흐름) 설계·적용 담당.

임팩트 iOS 팀 제품 저장소 7개 이상에 GitLab CI/CD 파이프라인 운영 / 병합 요청 시 자동 빌드·코드 검사·단위 테스트(제한적) / GitLab 병합 요청 리뷰봇 연계 / 2023.05 이후 운영 중

기술 Shell GitLab CI/CD

2022.04 - 진행 중 · 4년 3개월

### AppSuitSign 재서명 도구

iOS / macOS 통합 재서명 자동화 도구 AppSuitSign 을 4년째 운영하며 기능을 확장해 온 트랙

역할 AppSuitSign 도구 자체의 4년(2022.04-) 운영·기능 확장 주도 — iOS / macOS 통합 재서명 파이프라인 장기 유지보수, 고객 이슈 대응 및 Toolbox 앞단과의 결합.

임팩트 AppSuitSign 도구 자체의 4년(2022.04-) 운영 주도 / iOS .ipa + macOS .pkg + .app 단일 재서명 도구로 통합 / Objective-C 레거시 → Swift·SwiftUI 리팩토링 진행 / Developer ID Application 서명 키 운영 / Toolbox 앞단 결합으로 고객 SE 단일 워크플로우 정착

기술 Python3 Shell Objective-C Code Signing

2021.12 - 2022.04 · 5개월

### Python 2 → 3 마이그레이션

iOS 팀이 쓰던 Python 도구(빌드 후처리 패치 도구·AppSuitSign·자동화 스크립트)를 Python 2에서 3으로 전환하는 작업을 주도

역할 iOS 팀이 사용하던 Python 기반 도구(빌드 후처리 패치 도구, AppSuitSign, 자동화 스크립트)를 Python 2에서 3으로 옮기는 작업을 본인이 주도해 진행.

임팩트 iOS 팀 Python 도구 전체를 Python 2 → 3으로 전환(2021.12~2022.04) / 바이너리(Mach-O) 후처리에 쓰던 lief 라이브러리가 Python 3에서 바뀐 사용법을 반영해 도구 정상 동작 확보 / Apple Silicon(arm64) 환경의 Python 3에서 동작 검증 완료 / 더는 지원되지 않는 Python 2 의존을 끊어 도구의 장기 유지보수성 확보

기술 Python3

2021.10 - 진행 중 · 4년 9개월

### CLI\_XcodePlugin (재서명·후처리 자동화 플러그인)

AppSuit Premium 배포의 후처리·재서명을 자동화하는 CLI\_XcodePlugin 설계·유지·호환 테스트 협업

역할 AppSuit Premium 배포 워크플로우의 후처리 자동화 플러그인 CLI\_XcodePlugin 의 설계·유지·버전 호환 테스트 협업 (iOS 팀).

임팩트 Premium 배포 워크플로우 마지막 단계(재서명 자동화) 담당 / 2026-05-14 업데이트로 Premium 최신 메이저 릴리즈 호환 테스트 완료 / Cloud→Premium 명칭 변경에 따른 플러그인 가이드 정비

기술 Python3 Shell Code Signing

### StealPlate (앱스토어 심사 테스트용 iOS 앱)

AppSuit 적용 앱의 App Store 심사 통과 안전성을 증명하는 심사 테스트용 iOS 앱 StealPlate 설계·운영

역할 AppSuit 적용 결과의 App Store 심사 안전성 증명을 위한 심사 테스트용 iOS 앱 'StealPlate'의 설계·운영 담당. Premium·Air 신규 기능 출시 시 사전 검증 채널로 운영.

임팩트 App Store 등록·심사 테스트 통과 / AppSuit 적용 결과의 App Store 심사 안전성 증명 / Premium·Air 신규 기능 출시 시 사전 검증 앱으로 재사용 / iOS 팀의 Apple 심사 대응 도구로 정착

기술 Swift Objective-C XCFramework

## [PoC·기획]

2025.10 - 2026.05 · 8개월

### 보이스피싱 탐지 PoC (기본 통화 → WhatsApp/Telegram 탐지 확장)

해외 Premium 고객의 보이스피싱 통화 탐지 요구에 맞춰, 1차 PoC로 iOS 기본 통화 상태 탐지를 증명하고 2차 PoC로 WhatsApp-Telegram VoIP 통화 탐지를 검증

역할 iOS 보이스피싱 탐지 PoC 기획·구현·증명 / 2차 PoC(WhatsApp-Telegram VoIP 통화 탐지) 담당

임팩트 1차 PoC(2025.10.22~10.31): iOS 기본 통화에서 세션 ID와 통화 5상태(통화 안함·오고 있음·걸고 있음·통화 중·통화 보류 중) 관찰 증명, 앱별 샌드박스 한계(타 앱 통화 직접 관찰 불가) 명확화 / 2차 PoC(2026.05.18~05.24): WhatsApp-Telegram VoIP 통화 탐지 가능 여부 검증

기술 Swift XCFramework

2025.09 - 2026.05 · 9개월

### MAST (Mobile App Security Test) — 모바일 앱 취약점 점검 자동화 솔루션 기획

금융보안원 평가 항목 19개 + 자동화 시나리오 기반 자체 개념 검증(PoC) 설계 — 동적 진단(DAST) + 보고서 자동 출력, GUI/CLI 투트랙 1인 기획

역할 신규 개념 검증(PoC) 제품 '모바일 앱 취약점 점검 자동화 솔루션(MAST)'의 구상·기획·경쟁 제품 분석·일정 산정을 1인 단독 수행 — iOS 앱 보안 강화 운영 경험을 바탕으로 제품 제공 방식과 진단 항목 매트릭스 설계.

임팩트 금융보안원 평가 항목 19개 + 자체 추가 시나리오를 진단 항목 매트릭스로 1인 구조화 / 경쟁 제품 대비 제공 기능 매트릭스 작성 / GUI(macOS·Windows)·CLI 2종 제공 방식 설계(input: IPA, output: 진단 보고서) / 제품화 약 6개월 일정 산정 워크시트 완료.

기술 Swift Python3 Frida Libimobiledevice Binary Instrumentation

2024.02 - 2024.08 · 7개월

### iOS 매크로 탐지 솔루션 (MacroBlock) — 시장·타사 조사 후 개발 보류

iOS 매크로 탐지 신규 제품을 기획하고, OS 제약상 지원 가능한 탐지가 스위치 제어 검증 수준뿐이라는 근거로 개발 보류를 결정

역할 iOS 매크로 탐지 신규 제품 기획 리드 — 시장 조사·타사 솔루션 조사·기능 명세·개발 기간 산정 수행. "iOS OS 제약상 실효 탐지 기능이 스위치 제어 검증 하나뿐"의 한계를 근거로 제품화 보류 의사결정 제시·인수인계.

임팩트 기간 2024.02~08(기획·조사·보류 결정). 경쟁사 5개사+ 매트릭스(LIAPP·NHN AppGuard·appsealing·GxShield(AOS 전용)·Norton 등) / iOS 자체 매크로 부재 논거 4항목 정리 / 기능 명세·개발 기간 산정(2024.04.25~04.30) / 인수인계(2024.06.26~06.28) / 개발 보류 결정(2024.08.05).

기술 Swift Objective-C XCFramework

2023.04 - 2023.05 · 2개월

### Xcode Cloud CI/CD 통합 가능성 연구

Xcode Cloud CI/CD와 AppSuit 제품 파이프라인의 통합 가능성을 조사해 한계를 확인하고 보류 결론을 내린 연구

역할 Apple Xcode Cloud CI/CD와 AppSuit 제품군 통합 가능성을 조사·검토. 빌드 후처리 패치 단계의 수용 가능성을 분석하고 현 구조 적용 한계를 정리해 장기 보류 결론 도출.

임팩트 2023년 4~5월 조사. AppSuit 빌드 후처리 패치 단계(Mach-O 패치)의 Xcode Cloud 빌드 후 적용 가능성을 검토해 현 구조 적용 불가 확인, 장기 검토로 보류 결론.

## [팀 리딩]

2026.05 - 진행 중 · 2개월

### iOS 팀 회고 시스템 구축 (2026.05~)

iOS 팀 회고 시스템 착수·운영 주도 — 성과·이슈·개선·학습·감사 5축 보드로 구조화

역할 iOS 팀 월간·분기 회고 시스템의 설계·운영 주도 — 성과·이슈·개선·학습·감사 5축 보드로 구조화하고 팀의 자율 참여·논의 체계 수립.

임팩트 iOS 팀 회고 시스템 본인 추진 착수(2026.05) — 회고 보드 5축(성과·이슈·개선·학습·감사) 구조화, iOS 팀 자율 참여 구조, 월간·분기·연간 주기 확장 계획.

2025.07 – 진행 중 · 1년

### 솔루션사업본부 정기 회의·리더 파트 회의·본부 회고

iOS 팀장으로서 본부·리더 계층 회의에 지속 참석, 제품 현황 대변·의사결정 협의

역할 iOS 팀장으로서 솔루션사업본부 정기 회의·리더 파트 회의·본부 회고에 참석, iOS 팀 제품 현황·이슈를 상위 조직에 대변하고 제품 방향성 협의 협의.

임팩트 월간 본부 정기 회의·리더 파트 회의·분기 본부 회고 참석 / iOS 팀 이슈 상위 조직 대변 / 제품 방향성(MAST·AiSuit) 상위 조직 협의 채널 운영

2025.07 – 진행 중 · 1년

### iOS 차기 제품 후보 6개 트랙 동시 기획 (MAST·AiSuit 등)

iOS 팀의 다음 제품 후보 6개 트랙(MAST·AiSuit·보이스피싱·탐지 정책 용어 통일·서버 인증·옵션 UX)을 팀장으로서 동시에 기획하고 일정·완성도를 병행 관리

역할 iOS 팀장으로서 다수 제품 트랙(MAST / 옵션 체계 / 탐지 정책 비교 / 서버 인증 / 보이스피싱 / AiSuit)의 제품 방향성·명세·로드맵을 동시 설계·제안 주도. 단일 트랙 명세 중심에서 다수 트랙 병행 관리 체계로의 전환 설계.

임팩트 제품 후보 6개 트랙 동시 설계·관리 / MAST(앱 취약점 점검) 기획 완성 / AiSuit(ONNX 모델 보호) 개념 검증(PoC) 완성 / 보이스피싱 탐지 PoC 1차 완료·2차 진행 / 흩어져 있던 탐지 정책 4종 한국어 의미와 함께 용어 통일 / AppSuit Premium 옵션 UX 단계별 적용 계획 수립 / Premium 제품 관리 보드 구조화

2025.07 – 진행 중 · 1년

### iOS 팀 운영 (배정 체계 / 로드맵 70:30 / Notion·Jira 정비)

2025.07 팀장 승격 후 iOS 보안 솔루션 팀 운영 체계(배정·로드맵·도구)를 설계·운영 주도

역할 2025.07 팀장 승격 이후 iOS 팀 운영을 주도. 업무 배정 체계와 70:30(운영:신규) 로드맵, Notion·Jira 구조를 설계하고 주간·월간 운영 루틴을 주관. 설계와 구현을 병행하는 플레이어형 팀장.

임팩트 2025.07 팀장 승격 / iOS 팀 목표·업무 분장 문서 본인 주도 작성 / 70:30(운영:신규) 로드맵 고정·이행 / Notion·Jira 구조 정비 주도 / 연 2~3회 메이저 릴리즈 안정 수행

2025.07 – 진행 중 · 1년

### QA 체계 구축 (QA Board / Jenkins 자동 QA)

iOS 팀 QA 체계(QA Board + Jenkins 자동 QA 파이프라인)를 제품 메이저 릴리즈와 연동해 구축

역할 iOS 팀 QA 체계 구축 주도. Premium QA Board 재설계(본인 단독)와 Jenkins 기반 자동 QA 파이프라인 설계·연동 수행.

임팩트 QA Board 재설계 본인 단독 수행(2026-02-02~06) / 제품 × 차수 × 고객 적용 매트릭스 구조로 단일 보드에서 차수별 QA 현황 일괄 파악 / Jenkins 자동 QA로 제품별 샘플 앱 빌드·설치·탐지 검증 자동화 / 2026년 1월·5월 메이저 릴리즈에 QA Board·자동 QA 연동 적용

기술 Python3 Shell Jenkins Pipeline

2025.03 – 진행 중 · 1년 4개월

### iOS 팀 개발 프로세스 정비 (Git Flow / MR / 코드 리뷰 표준)

iOS 팀 개발 프로세스 — Git Flow 브랜치 전략·MR 템플릿·코드 리뷰 규칙 — 수립·정비 주도

역할 iOS 팀 개발 프로세스 설계·정비 단독 주도 — Git Flow 기반 브랜치 전략, MR 템플릿, 코드 리뷰 규칙, 고객사 커스텀 패치 추적 표준 수립

임팩트 2025-03 iOS 팀 회의 'MR 피드백 조율' 안건으로 표준화 착수 / MR 템플릿 4축(제목·요약·리뷰 포인트·참조) 정착 / CodeReview Guide 문서화(24시간 이내 리뷰 원칙, 리뷰 대상 제품 7종) / develop 머지=히스토리 유지·release 머지=squash 규칙 수립 / Jira 이슈 연동 / Premium 여러 릴리즈 사이클에 적용

기술 GitLab CI/CD

2025.01 – 2025.12 · 1년

## iOS 팀 인수인계 주도 (메인·서브·AVSuit·SaaS Air 순)

1인 축적 iOS 제품군을 새 팀원에게 이관하는 계단식 인수인계 순서(메인→서브→AVSuit→SaaS Air)를 설계·주도

역할 iOS 팀 확장기에 4개 제품군의 인수인계 순서·교육 자료·이관 검증 기준을 단독 설계하고 이관 전 과정을 주도

임팩트 2025.01~04 1차 인수인계 완료(메인·서브·AVSuit·SaaS Air 순) / 4개 제품군별 이관 검증 자료 단독 설계 / iOS 팀을 팀원 각자 독립 운영하는 구조로 강화 / 2025.12 SaaS Air 최종 인수로 마무리

## [사내 인프라]

2026.04 - 진행 중 · 3개월

### iOS 팀 GitLab MR 리뷰봇 (TextRAG 기반 사내 코드 리뷰 자동화)

사내 GitLab MR 이벤트를 받아 TextRAG의 RAG 코드 리뷰 API를 호출·자동 의견을 게시하는 사내 리뷰봇을 구축

역할 사내 GitLab MR에 대해 TextRAG의 RAG 기반 코드 리뷰 API를 호출, 자동 리뷰 의견을 MR 코멘트로 게시하는 리뷰봇의 설계·개발·운영을 담당. 기존 TextRAG 서버를 코어 엔진으로 재사용하는 통합 설계 담당.

임팩트 2026-04-20~04-22 구현. TextRAG의 RAG 코드 리뷰 API를 코어 엔진으로 재사용. iOS 팀 7개 제품(Premium / Air / VPNBlock / AVSuit 등) MR에 2026-04-22부터 자동 리뷰 코멘트 적용 / 버그·보안·성능·품질·정합성 5축 리뷰

기술 Python3 FastAPI Ollama GitLab CI/CD

2026.02 - 진행 중 · 5개월

### AI 개발 워크플로우 엔지니어링 패턴 도입 (사내 LLM 기반 개발 워크플로우)

보안 제품사 외부 LLM 금지 제약 아래, 사내 LLM 기반 개발 워크플로우 패턴을 설계·도입 주도

역할 사내 LLM 기반 개발 워크플로우(TextRAG·Claude Code·GitLab MR 리뷰봇·개인 OSS 인사이트)를 'AI 개발 워크플로우 엔지니어링' 패턴으로 구조화·사내 적용 주도. 패턴 설계 후 실적용은 iOS 팀 단위로 확장.

임팩트 TextRAG·GitLab MR 리뷰봇 사내 운영. 설계·구현·리뷰·문서화 각 단계에 LLM 점진 적용 패턴 수립. 경력 DB·이력서·자소서 자동화에도 동일 구조 적용. book-forge(도서 원고→docx)·Glimi 등 개인 OSS 경험을 패턴으로 추상화.

기술 Python3 FastAPI Ollama vLLM

2026.02 - 진행 중 · 5개월

### 사내 RAG 시스템(TextRAG) 도입·운영

외부 LLM에 사내 자료를 보낼 수 없는 제약 아래, 로컬 LLM·벡터 DB 기반 사내 RAG를 설계·구축해 운영하고 iOS 팀 도입까지 진행

역할 보안 제품사의 외부 LLM 금지 제약 아래 로컬 RAG 시스템을 구축. 아키텍처 설계, MCP 검색 도구 구현, 메모리 예산 시나리오 설계, iOS 팀 도입 가이드 작성·교육 담당.

임팩트 로컬 RAG 서버 이원 구조(인덱싱 전용 + 주 서버) / 15개 코퍼스 상주(iOS 팀 코드베이스 7종 + 문서 코퍼스 2종 7천 청크 이상 + 보조 코퍼스) / Claude Code MCP 검색 도구 노출 완료 / GitLab MR 리뷰봇 코어 엔진으로 재사용 / 2026.02부터 운영

기술 Python3 Shell FastAPI Ollama vLLM

## 개인 프로젝트

2026.06 - 진행 중

### 이력서 RAG 챗봇 — 로컬 LLM 직접 운영

사이트의 모든 공개 데이터를 검색해 심재빈 1인칭으로 답하는 챗봇으로, 로컬 서버에서 검색부터 생성까지 전 구간을 직접 운영한다.

임팩트 외부 AI API 없이 로컬 서버에서 직접 운영해 운영비 0원 · 2단 검색으로 관련 자료만 근거로 사용 · 질문과 무관한 출처가 뜨던 문제 제거

2026.05 - 진행 중

### 노션 기반 자동 동기화 포트폴리오 사이트 (jempty.iruyo.com)

노션을 source of truth로 두고 매시간 자동 동기화되며, 빌드·배포·PDF 생성까지 전부 자동으로 따라오는 이력서·포트폴리오 사이트

2025.12 – 2026.01

### IntentCP — Smart Home Intent Control Plane (Python/FastAPI)

집에 GPU-LLM 서버 없이도 '저녁 분위기 만들어줘' 같은 말 한마디를 iOS 단축어-Siri로 스마트홈 제어로 바꿔주는 중간 서버(control plane)를 설계·구현해 GitHub 공개

임팩트 GitHub 공개 오픈소스(github.com/jaebinsim/IntentCP) — 유튜브 채널 영상 노출 이후 22 stars 확보(2026.05 기준). 본인 자취 환경 자동화 도구로 직접 운영하며 지속 개선.

---

2025.11 – 2026.02

### ShortKit — iBeacon Proximity Webhook Trigger (iOS 앱)

비콘이 가까워지면 미리 정한 주소를 자동 호출하는 iOS 개인 앱, 실내 근접 자동화 OSS를 1인 구축

임팩트 GitHub 공개 OSS(github.com/jaebinsim/ShortKit) / iOS-Swift / 2025.11.20 시작·운영 중 / 감지 앱(ShortKit)과 실행 앱(IntentCP)을 짝지어 '비콘 감지 → 주소 호출 → 스마트홈 제어' 한 벌을 1인 구성 / 유튜브 자취남 채널 영상(2026.01.30) 자동화 시연에 사용

---

2025.01 – 진행 중

### Glimi — Multi-agent Harness OSS (proactive supervisor 설계)

사람이 자리를 비워도 AI 에이전트들이 자기들끼리 대화·관계를 이어가는 멀티 에이전트 오픈소스 Glimi를 설계·구현

임팩트 1인 개발 · 2026 하반기 오픈소스 공개 준비 중 / LLM 호출 하나를 8개 계층(프롬프트·도구·기억·채널·가드·에이전트 간 대화·자가복구·능동 관리자)으로 감싸 에이전트 군집 운영 / 클라우드(Claude)·로컬 모델(Ollama·vLLM·llama.cpp)을 재시작 없이 교체 / 5단 영속 기억으로 관계·사실 장기 보존 / 에이전트 상태 실시간 대시보드 / 설계 문서 13종 · pip 첫 배포 준비 중

---

2019.04

### C언어 최단거리 알고리즘 (데이크스트라, 2019)

데이크스트라 알고리즘 기반 가중 그래프 최단거리 C 프로그램 (2019, 고등학교 시절)

임팩트 C / 데이크스트라 알고리즘 / 가중 그래프 최단경로 산출 (2019).

---

2018.01 – 2018.06

### ASUS 공유기를 이용한 NAS 서버 구축 (2018)

ASUS 공유기를 NAS 서버로 전환해 Transmission-TVheadend·웹 서버를 구축한 고등학교 시절 개인 프로젝트

임팩트 2018.01~06 개인 프로젝트 / Entware 기반 패키지 설치 / Transmission-TVheadend·웹 서버 3종 구축 / 고등학교 시절(만 16세) 임베디드·리눅스 환경 자가 학습

---

2018.01 – 2019.12

### Google Cloud 게임 서버 구축 (2017~2020)

Google Cloud Platform 기반 개인 게임 서버 구축·운영 (고등학교 시절 개인 프로젝트)

임팩트 고등학교 시절 클라우드(GCP) 기반 서버 구축·운영 경험 / 서버 명세 작성 경험

---

2017.02 – 2017.11

### 이루요 (iruyo.com) 반응형 커뮤니티 웹사이트 구축 (2017)

Apache·MariaDB·XpressEngine 기반 독립 도메인 iruyo.com 반응형 커뮤니티 웹사이트 구축 (2017, 고등학교 시절)

임팩트 고등학교 재학 시절(만 15세) 개인 웹 구축 경험 / CentOS·Apache·MariaDB·PHP·XpressEngine 풀스택 / iruyo.com 독립 도메인 운영 / 2017.02~11 (약 9개월)

---

## 특허

---

2023.06 - 2023.12

특허 출원 — 텍스트 필드 보호 속성을 이용한 스크린 캡처 및 미러링 방지 장치 및 방법

(주)스틸리언

iOS UIView / Layer 계층 조작 + 텍스트 필드 보호 속성 확장으로 스크린 캡처·미러링을 차단하는 기술

특허 공동 출원 1건 (출원인 □스틸리언, 공동발명자)

## 출판·집필

---

2026.06

『비개발자를 위한 클로드 활용 가이드』 단독 저자

코리아교육그룹

Claude 를 코딩 없이 업무에 녹여 쓰는 패턴을 비개발자 독자 대상으로 정리한 단독 저서. 2026.06 출간 예정.

단독 집필 · 출판 계약 완료 (2026.06 출간 예정)

## 병역

---

2022.01 - 2023.12

산업기능요원 복무 완료

재직 중 병행 복무, 2023.12 만료