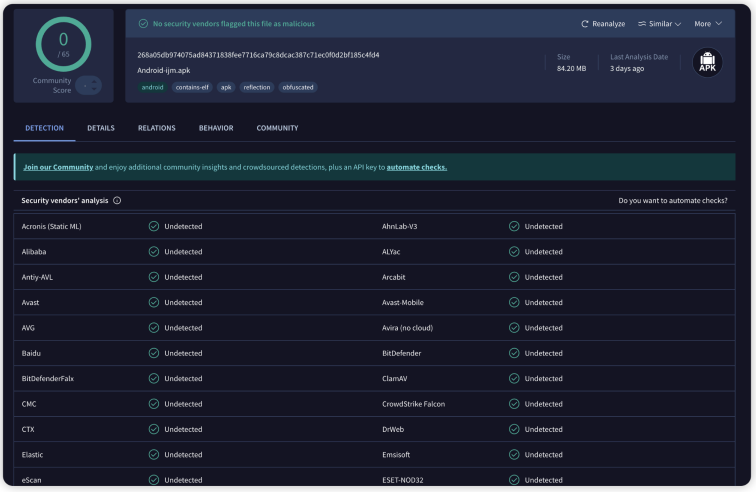
项目概述

项目简介

SafeW是一款高度安全的即时通讯应用，专为那些将隐私和安全放在首位的用户和企业设计，支持私有化部署，借助基于Telegram端到端加密技术，SafeW确保您的对话内容在传输中完全加密，确保只有对话双方能够访问信息内容。

版本信息

- **平台**：Android
- **下载链接**：https://safew.org/assets/Android.apk
- **版本信息**：v1.8.6
- **校验信息**：
  - MD5: fda2798f322d8941e7d84ea1726816b7
  - SHA256: 268a05db974075ad84371838fee7716ca79c8dcac387c71ec0f0d2bf185c4fd4
- **安全检测**：
  - **腾讯移动安全实验室检测**：通过18项核心检测（包括渗透测试、漏洞扫描等）
  - **VirusTotal 检测报告**：65款杀毒引擎全绿通过



应用安全

安全合规性与认证

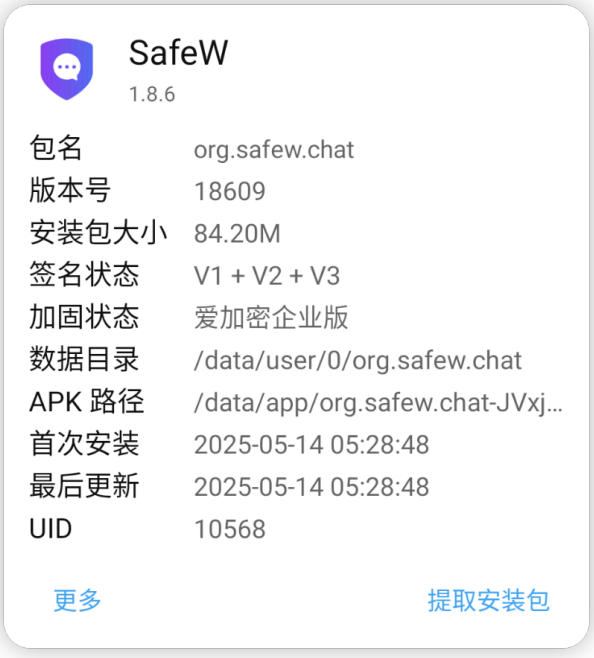
SafeW 在设计与开发过程中严格遵循以下国际安全标准：

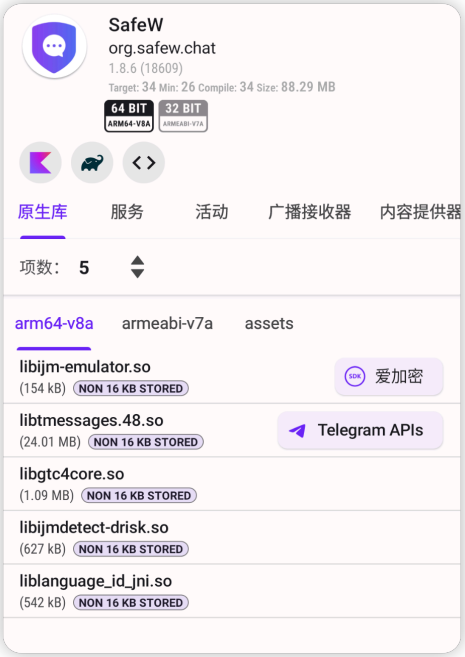
- ISO/IEC 27001：信息安全管理体系认证
- ISO/IEC 27034：应用安全生命周期认证
- OWASP Mobile Top 10：移动应用十大安全风险防范标准
- GDPR / 网络安全法：符合隐私合规要求（如用户知情、数据可控、可撤回等）

应用安全防护措施

SafeW 采用了多层次、多维度的安全加固策略，从反编译防护、恶意分析防护到签名完整性校验，全面提升 App 的安全性。

- 应用层加固
  - 基于爱加密企业级加固方案，实现DEX文件VMP虚拟化保护，有效对抗静态逆向分析
  - 代码混淆强度达到商业级标准（控制流扁平化+字符串加密+反射调用混淆）
  - 完整性校验采用双因子验证机制（APK签名校验+完整性校验）
- 运行时防护
  - 动态环境检测模块实时监控Root环境、调试器附加、Frida等注入行为
  - 基于Hook框架特征码的主动防御机制（Xposed/Frida检测准确率>99.8%）
  - 模拟器识别覆盖主流平台（BlueStacks/MuMu/夜神等），触发后启动逻辑自毁流程
- 应用使用的第三方 SDK 均来源合法，且经过定期安全评估与更新：
  - libtmessages.48.so - Telegram 核心通信 API
  - libgtc4core.so - 极验 4.0 验证模块
  - liblanguage_id_jni.so - Google 语言识别模块
  - libijm-emulator.so / libijmdetect-drisk.so - 爱加密安全模块

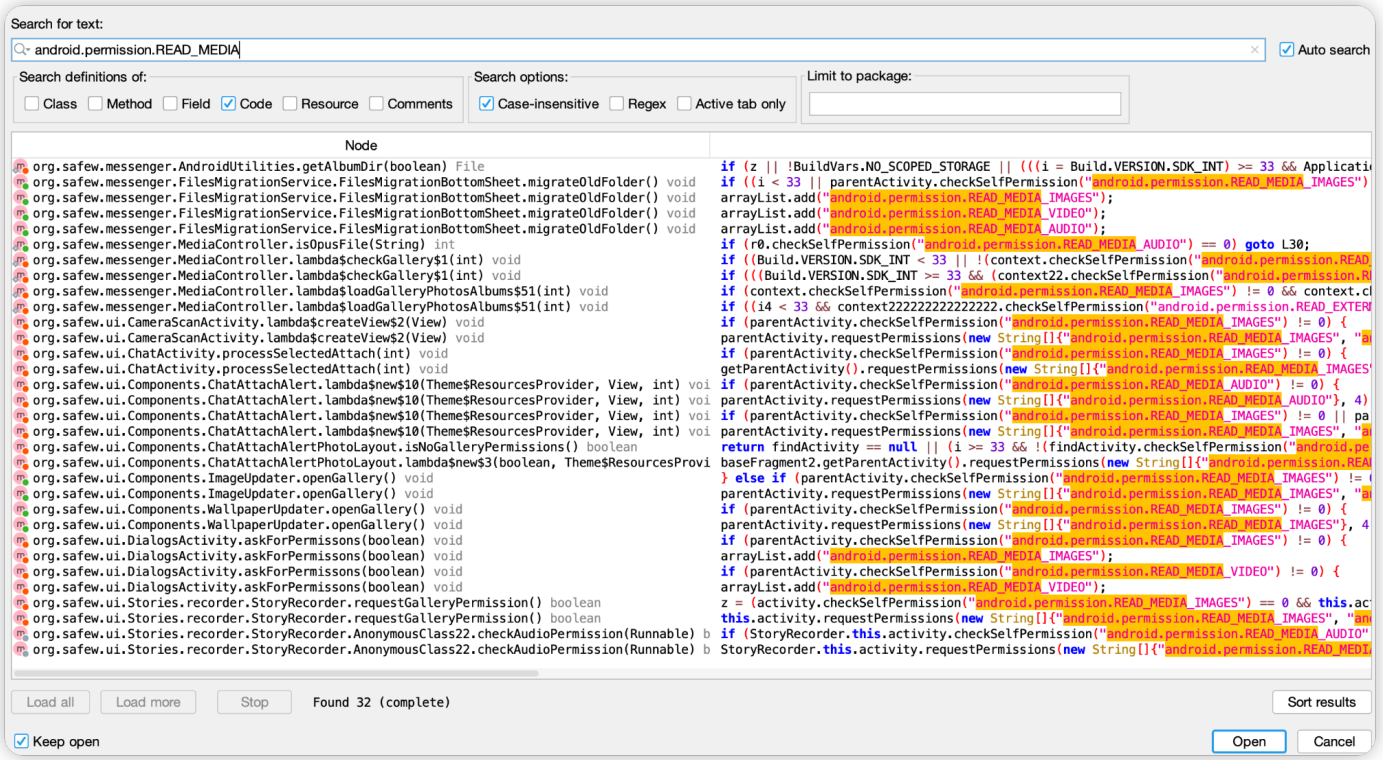




隐私保护机制

SafeW 致力于保护用户的个人隐私数据，遵循国家相关法律法规及国际隐私保护标准：

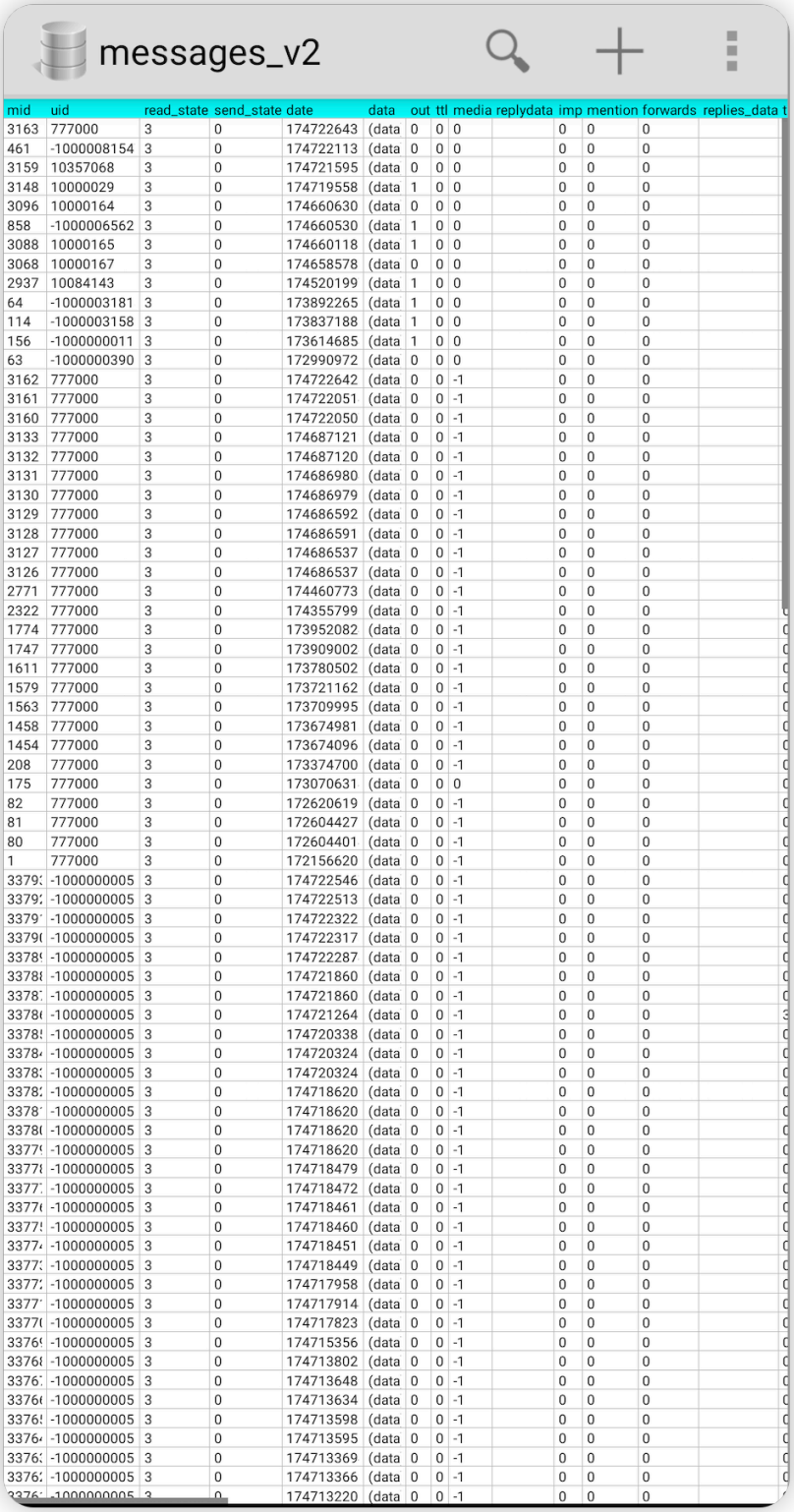
- **最小权限原则**：
  - 所有权限均在运行时按需申请，并明确告知用户使用目的。
  - 用户可自由控制是否授权，避免不必要的权限泄露。
- **隐私告知机制**：
  - 应用在首次启动及隐私政策中明确列出收集哪些信息、用途及保护措施。
  - 涵盖联系人、位置信息、设备信息等敏感数据的处理方式，保障用户知情权。
- **数据删除与导出功能**：
  - 用户可以随时清除聊天记录、本地缓存与账号信息，应用提供完整的“数据清理”与“隐私导出”接口。
- **权限请求设计合理**：
  - 应用在代码层面对权限管理进行了精细设计，确保仅在用户进行相关操作、确有实际业务需求时才发起权限申请。
  - 权限调用逻辑清晰、结构可控，最大限度减少对用户隐私的打扰与误用风险。



数据安全机制

SafeW 对本地敏感数据（如聊天消息、用户资料、媒体索引等）进行了全面加密处理：

- 所有敏感数据统一存储在 cache4.db 本地数据库中，并通过 SafeW 原生加密机制加密。
- 加密使用的密钥并不以明文形式存储，而是基于用户设备信息、登录会话动态生成并临时加载至内存中。
- 加密算法为 AES-256，结合 SafeW 的加密模块形成封闭式保护体系，防止数据泄露与伪造。



messages_v2

mid	uid	read_state	send_state	date	data	out	ttl	media	replydata	imp	mention	forwards	replies_data
3163	777000	3	0	174722643	(data	0	0	0		0	0	0	
461	-1000008154	3	0	174722113	(data	0	0	0		0	0	0	
3159	10357068	3	0	174721595	(data	0	0	0		0	0	0	
3148	10000029	3	0	174719558	(data	1	0	0		0	0	0	
3096	10000164	3	0	174660630	(data	0	0	0		0	0	0	
858	-1000006562	3	0	174660530	(data	1	0	0		0	0	0	
3088	10000165	3	0	174660118	(data	1	0	0		0	0	0	
3068	10000167	3	0	174658578	(data	0	0	0		0	0	0	
2937	10084143	3	0	174520199	(data	1	0	0		0	0	0	
64	-1000003181	3	0	173892265	(data	1	0	0		0	0	0	
114	-1000003158	3	0	173837188	(data	1	0	0		0	0	0	
156	-1000000011	3	0	173614685	(data	1	0	0		0	0	0	
63	-1000000390	3	0	172990972	(data	0	0	0		0	0	0	
3162	777000	3	0	174722642	(data	0	0	-1		0	0	0	
3161	777000	3	0	174722051	(data	0	0	-1		0	0	0	
3160	777000	3	0	174722050	(data	0	0	-1		0	0	0	
3133	777000	3	0	174687121	(data	0	0	-1		0	0	0	
3132	777000	3	0	174687120	(data	0	0	-1		0	0	0	
3131	777000	3	0	174686980	(data	0	0	-1		0	0	0	
3130	777000	3	0	174686979	(data	0	0	-1		0	0	0	
3129	777000	3	0	174686592	(data	0	0	-1		0	0	0	
3128	777000	3	0	174686591	(data	0	0	-1		0	0	0	
3127	777000	3	0	174686537	(data	0	0	-1		0	0	0	
3126	777000	3	0	174686537	(data	0	0	-1		0	0	0	
2771	777000	3	0	174460773	(data	0	0	-1		0	0	0	
2322	777000	3	0	174355799	(data	0	0	-1		0	0	0	
1774	777000	3	0	173952082	(data	0	0	-1		0	0	0	
1747	777000	3	0	173909002	(data	0	0	-1		0	0	0	
1611	777000	3	0	173780502	(data	0	0	-1		0	0	0	
1579	777000	3	0	173721162	(data	0	0	-1		0	0	0	
1563	777000	3	0	173709995	(data	0	0	-1		0	0	0	
1458	777000	3	0	173674981	(data	0	0	-1		0	0	0	
1454	777000	3	0	173674096	(data	0	0	-1		0	0	0	
208	777000	3	0	173374700	(data	0	0	-1		0	0	0	
175	777000	3	0	173070631	(data	0	0	0		0	0	0	
82	777000	3	0	172620619	(data	0	0	-1		0	0	0	
81	777000	3	0	172604427	(data	0	0	-1		0	0	0	
80	777000	3	0	172604401	(data	0	0	-1		0	0	0	
1	777000	3	0	172156620	(data	0	0	-1		0	0	0	
3379	-1000000005	3	0	174722546	(data	0	0	-1		0	0	0	
3379	-1000000005	3	0	174722513	(data	0	0	-1		0	0	0	
3379	-1000000005	3	0	174722322	(data	0	0	-1		0	0	0	
3379	-1000000005	3	0	174722317	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174722287	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174721860	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174721860	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174721264	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174720338	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174720324	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174720324	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174718620	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174718620	(data	0	0	-1		0	0	0	
3378	-1000000005	3	0	174718620	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718620	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718479	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718472	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718461	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718460	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718451	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174718449	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174717958	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174717914	(data	0	0	-1		0	0	0	
3377	-1000000005	3	0	174717823	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174715356	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713802	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713648	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713634	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713598	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713595	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713369	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713366	(data	0	0	-1		0	0	0	
3376	-1000000005	3	0	174713220	(data	0	0	-1		0	0	0	

通信加密安全

SafeW 的核心通信加密机制基于 Telegram 的 **MTProto 2.0 协议**，具备以下安全特性：

MTProto 2.0 优势

1. **抵御已知攻击**：
   - 禁止 Diffie-Hellman 密钥复用，防止中间人攻击。
   - 强化 MAC 消息认证机制，保障数据完整性与防篡改。
   - 引入动态 salt + 改进随机数生成机制，提升对抗重放攻击能力。
2. **结构模糊化处理**：
   - 消息内容经过填充（padding）与混淆，降低流量分析与模式识别攻击风险。
3. **前向安全性（Forward Secrecy）**：
   - 每次会话使用独立密钥，历史通信不会因未来密钥泄露而受影响。
4. **双向身份验证机制**：
   - 通信双方均需参与密钥协商，杜绝身份伪造。

E2EE（端到端加密）特性

1. **密钥仅存于通信双方设备**：
   - 所有 E2EE 聊天使用本地生成密钥，不上传服务器。
   - 即使 SafeW 服务器被攻破，也无法解密用户消息。
2. **聊天不可跨设备同步**：
   - 安全性优先设计，秘密聊天记录仅保留于当前设备。
   - 更换设备或卸载应用后，需重新建立新会话。
3. **抵抗取证与监听攻击**：
   - 本地数据库采用高级别 AES 加密，调试接口默认关闭，配合反调试机制与行为检测系统，有效阻止运行时内存注入和数据外泄

抗抓包设计

- 所有通信均通过 MTProto 2.0 协议 而非标准 HTTPS 实现，其数据包结构并不符合通用协议特征，无法被常规抓包工具（如 Charles、Fiddler、Burp Suite）识别和解码。
- 客户端实现了连接安全性校验、证书校验以及数据内容加密混淆，即使在中间代理拦截下，抓到的数据也不可解密分析。
- 加之对代理、VPN、调试工具的识别能力，进一步提升了抗嗅探能力。