OpenPGP 复合 KEM 加密消息的安全性分析

本报告的其他语言版本：English。

rfc9980 · ProVerif+ Agent 报告

文稿	Post-Quantum Cryptography in OpenPGP （协议定义）; OpenPGP （辅助）
检查的性质	12 条：11 条成立，1 条不成立（1 个攻击）
发现	1

目录

1 协议和范围
2 发现
3 检查过的性质
4 本次分析的局限
附录 A：逐条性质的解读
附录 C：主张和证明器的答案
附录 D：分析是怎么跑的

1 协议和范围

该协议将一条 OpenPGP 消息加密给一个接收者，该接收者的复合 ML-KEM-768 + X25519 公钥已知且可信。发送方对该密钥的两个分量分别封装，用 SHA3-256 将 ML-KEM 密钥份额、X25519 密钥份额、算法 ID、接收者的 ECDH 公钥和一个域分隔符组合成密钥加密密钥，并在其下封装会话密钥。v6 PKESK 包携带两个密文和被封装的密钥，消息本身则在 v2 SEIPD 包中传输，其消息密钥由会话密钥和一个新盐派生。

接收者持有两个私钥。它对两个分量分别解封装，用自己的公钥和所配置的算法 ID 重新计算密钥加密密钥，解封会话密钥，并在密钥封装完整性检查或 SEIPD 认证标签失败时停止。协议不保证对方是谁：任何持有接收者公钥的一方都可以向它构造消息，而发送方不会因此得知任何身份信息。

各方：

- **发送方**：将一条消息加密给一个可信公钥。持有：接收者的复合公钥：R（ECDH）和 ML-KEM 封装密钥；algId。
- **接收者**：持有复合方案的两个私钥。持有：r（ECDH 私钥）、ML-KEM 解封装密钥。

图 1 展示了文稿所规定的交换过程。

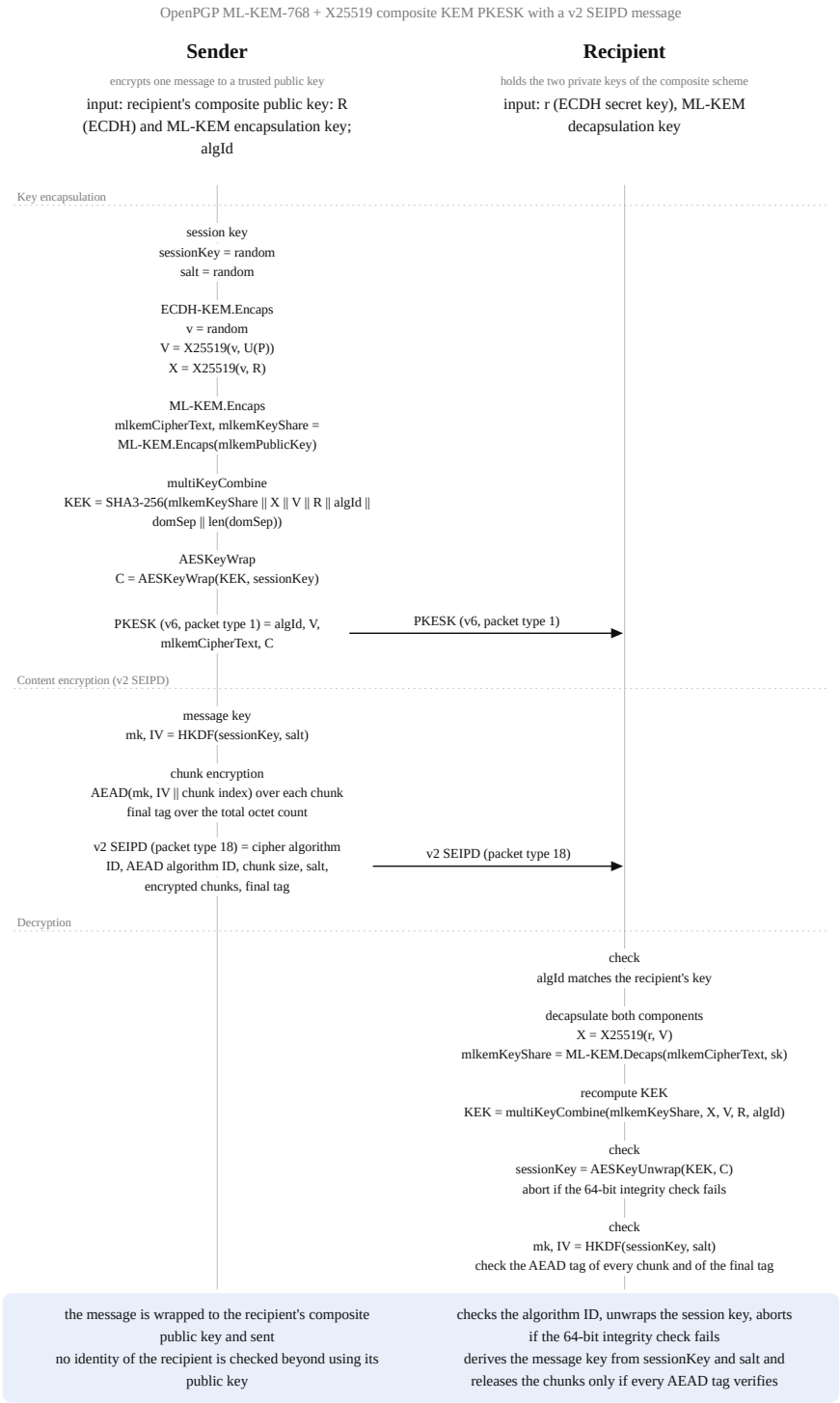


图 1. 文稿所规定的协议。

分析的范围见附录 D：分析了什么、省略了什么，以及它所依据的文稿章节。

2 发现

本次分析有 1 项发现：以下内容文稿中并未写明。其中 1 项，文稿是否写明不明确。

在所建模的范围内，该协议提供了它应当提供的保证：网络上的攻击者，以及攻破复合 KEM 中一个分量的攻击者，既得不到密钥加密密钥，也得不到会话密钥，更得不到明文。这一保护的边界是确切的：接收者的两个私钥同时泄露会打开每一条已记录的消息，因此不存在前向保密；而攻击者手中的 KEK 或会话密钥

只能打开它自己的那条消息。在十二个目标中，十一个得到证明，一个只对消息的三个展开块得到证明，还有一个因前向保密攻击而不成立。没有发现是新的：文稿写明了组合器的 IND-CCA2 论证、AEAD 完整性规则以及算法 ID 与公钥的绑定，模型复现了这些内容；文稿对前向保密只字未提，而这本就是加密到静态密钥所固有的。结果依赖于理想化的 SHA3-256、AES 密钥封装、ML-KEM 和 AEAD，以及只含交换律的 Diffie-Hellman，因此它们对这些原语的强度不置一词。

攻击图使用图 1 的列。红色是攻击者所做的操作。黄色标记某一方发送了指定交换在该位置本不应有的消息，以及被攻击破坏的结果。

2.1 泄露的接收者密钥可打开每一条已收到的消息

类型	协议规范本身的弱点
是否新发现	不确定：文稿只写了一部分，或没能检索
价值	低
涉及的性质	C01 （见第 3 节）
原文	s0064, s0084, s0085, s0097, s0098
依据	ProVerif 攻击查询 `c01`; 图 2

发生了什么。 接收者的长期私钥可解密它已经收到的每一条消息。攻击者用 ECDH 私钥从记录的密文重新计算 X25519 密钥份额，用 ML-KEM 解封装密钥解封装记录的 ML-KEM 密文；随后密钥组合器给出密钥加密密钥，密钥封装打开会话密钥，SEIPD 包中的盐给出消息密钥。

攻击者需要什么。 接收者的两个私钥，在消息收到之后获得；已记录的 PKESK 和 SEIPD 包，网络攻击者会保留它们。

攻击者能得到什么。 攻击者能读取接收者用该密钥曾经收到的每一条消息，以及接收者更换密钥之前将收到的每一条消息。没有任何信号提示密钥已丢失。

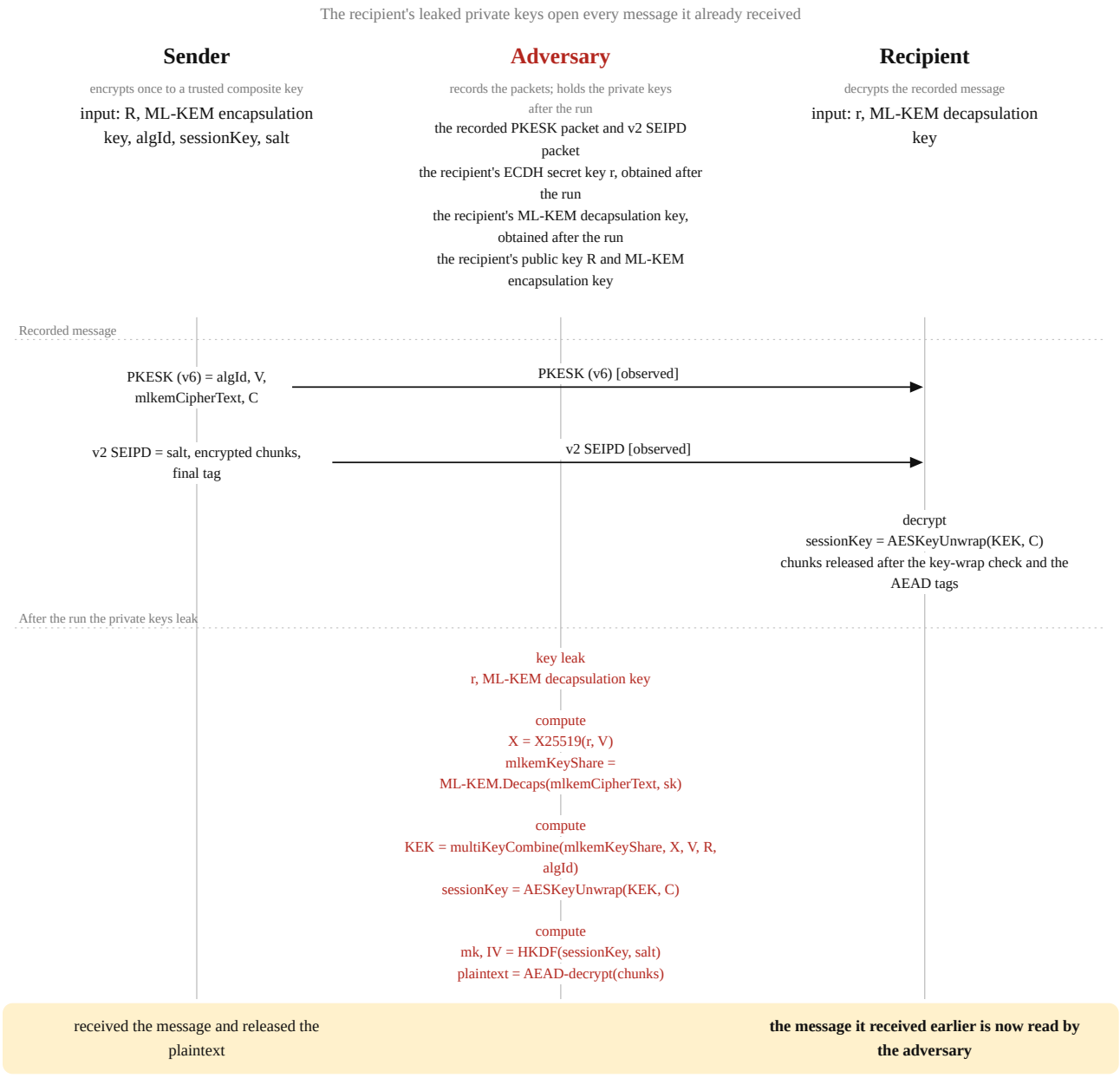


图 2. 接收者泄露的私钥可打开它已收到的每一条消息。性质 C01；攻击查询 c01。

拟议修改。 无。对消息或派生方式的任何改动都无法消除这一点：两个密钥份额都是接收者静态 ECDH 密钥和静态 ML-KEM 解封装密钥的函数，因此前向保密需要交互式或棘轮式交换，那是另一种协议，无法在本模型上测试。

3 检查过的性质

每一行是一条提交给 ProVerif 的性质。结果是 ProVerif 对分析者所建协议模型的回答；价值和文稿是否已写明该结果由分析者判断，理由见附录 A。

ID	性质	假定的泄露	结果	价值	文稿是否已写
K01	**KEK与私钥的保密性。** 攻击者不能推导出 KEK、会话密钥，也不能推导出接收者的 ECDH 私钥与 ML-KEM 私钥。	无	成立	中	是

ID	性质	假定的泄露	结果	价值	文稿是否已写
K02	**消息明文的机密性。** 消息明文只对预期接收者可见，网络攻击者不能得到明文。	无	成立	中	是
A03	**关于所接受KEK的一致性。** 接收者成功解出会话密钥时，该 KEK 由发送者针对接收者公钥、算法 ID 与本次封装密文计算得到。	无	成立	低	否
A08	**KEK绑定到算法与密钥。** 接收者接受的 KEK 所绑定的 algId、ecdhPublicKey、ecdhCipherText 与其自身密钥和收到的 PKESK 一致，攻击者不能把为其他算法或公钥生成的封装挪用到本接收者。	无	成立	中	是
C08	**单个分量被攻破时 KEK 仍保密。** 在 X25519 分量失效（攻击者得到 ecdhKeyShare）或 ML-KEM 分量失效（攻击者得到 mlkemKeyShare）之一的情况下，KEK 与会话密钥仍对攻击者保密。	复合 KEM 的单个分量（X25519 或 ML-KEM）(加密之后、解密之前或任意时刻)	成立	中	是
C01	**过去消息无前向保密。** 会话完成后泄露接收者长期私钥，攻击者仍不能得到此前会话的 KEK、会话密钥与明文。	接收者 (会话完成后泄露)	不成立（1 个攻击）	低	不确定
S01	**AEAD nonce不复用。** 同一消息密钥下，AEAD nonce（IV 与块索引组合）不重复使用。	无	成立	低	是
K01.8	**会话密钥与明文保密。** 攻击者不能得到会话密钥与消息明文。	无	成立	中	是
A08.9	**KEK绑定到algId、密钥、密文。** 接收者接受的 KEK 所绑定的 algId、ecdhPublicKey、ecdhCipherText 与其自身密钥和收到的 PKESK 一致。	无	成立	中	是
C08.10	**组件泄露韧性并含双泄露对照。** 在 X25519 分量泄露、ML-KEM 分量泄露或两者同时泄露的情况下，检查 KEK 与会话密钥是否仍保密；单分量泄露应保持，双分量同时泄露作为对照。	复合 KEM 的经典分量与/或 PQ 分量 (加密之后、解密之前或任意时刻)	成立	中	是
X01	**SEIPD明文完整性。** 接收者绝不会释放在该会话下并非从发送者处收到的明文：如果它从 v2 SEIPD 包中恢复出明文 m，那么发送者在同一会话中封装了 m，因此被修改或伪造的密文不会被当作新明文接受。	无	成立	中	是
X02	**逐消息KEK分离。** 两条消息绝不共享同一个 KEK：如果两次封装得到相同的 KEK，它们就是同一次封装，因此一条消息的 KEK 泄露对另一条消息毫无影响。	无	成立	低	否

文稿写明而起草的清单里没有、由分析者补充的目标（见附录 D）：X01, X02.

所提出的问题以及分析者的回答：

问题	回答	依据
Q1: 明文与会话密钥保密	会话密钥和明文对仅掌握网络攻击能力的攻击者、以及只攻破两个 KEM 分量之一的攻击者都不可及；只要接收者的两个长期私钥同时泄露，它们就会丢失。	K01.8, K01, K02, C01
Q2: 算法 ID/接收者公钥/封装密文对 KEK 的绑定	KEK 绑定到算法 ID、接收者的 ECDH 公钥以及 ECDH 和 ML-KEM 密文：在另一算法下或为另一方 ECDH 密钥所做的封装绝不会被接受，接收者为发送方封装所计算的 KEK 就是发送方的 KEK。	A08.9, A08, A03
Q3: 分别泄露经典与 PQ 分量时的理想组合器结论及双泄露控制	单独攻破任一分量都不会使 KEK 和会话密钥泄露，因此组合器的行为与文稿所述一致；两个分量同时被攻破时 KEK 会被立即推导出来，这正是两个证明的对照。	C08.10, C08

4 本次分析的局限

这里的结果是建立在协议模型上的结果：它说明的是所规定的消息和检查能提供什么、不能提供什么，其中密码学被当作理想的，且不涉及实现。

据分析者所述，本次分析未覆盖的内容如下：

- SHA3-256、AES 密钥封装、ML-KEM 与 AEAD 都是理想的构造器和解构器：不存在利用其结构的攻击，也不对其计算强度作任何陈述。
- Diffie-Hellman 只带有共享坐标对称这一条定律。该群没有阶、没有小阶点，也没有贡献性行为，因此 rfc7748:810 与 rfc7748:820 中关于小阶输入和等价公钥的 X25519 警告未被检验；建模该代数的 Tamarin+ Agent 可以解决这些问题。
- 长度、编码以及 PKESK 与 SEIPD 数据包的八位组布局均未建模；v3 PKESK 中“解封后的密钥长度与所声明的对称算法一致”这一检查（s0111）不在建模流程之内，因为 v2 SEIPD 数据包与 v6 PKESK 配对（s0147、s0155）。
- 证书、密钥分发以及 PKESK 中用于标识接收者的字段均不在范围内，因此接收者匿名性（s0200）和指纹检查未被分析。
- 建模了一个接收者密钥对、一个 ECDH 密钥和两个复合算法 ID；多接收者消息不在范围内，文稿指出除非每个密钥都是后量子安全的，否则其机密性不是后量子安全的（main:1133）。
- 主模型建模了一个分块，有界模型建模了三个；末尾分块的截断、已认证分块的流式释放以及最终长度标签均未建模。
- 随机生成器的强度无法建模：新名称被假定为不可猜测，因此文稿中的 CSPRNG 要求是假定而非检验的。
- 恶意发送者未被分析，因为任何一方都可以向公钥构造消息；发送方仅以诚实加密者的身份进入模型。
- 侧信道、时序以及通过消息长度进行的流量分析无法向证明器提出。

宿主记录的内容：

- S01 的答案依赖于模型简化的一项操作（附录 D）。

每条性质仍有哪些未决之处、以及什么能解决它，见附录 A。

附录 A：逐条性质的解读

这是分析者本人对证明器答案的解读，而非证明器的答案：这些答案对协议意味着什么、解读依据什么、能否照原样使用、以及它的价值何在。

K01: KEK与私钥的保密性

攻击者不能推导出 KEK、会话密钥，也不能推导出接收者的 ECDH 私钥与 ML-KEM 私钥。

看到每个数据包的攻击者无法推导出 KEK、会话密钥或接收者的任一私钥。KEK 需要两份密钥份额，而它们分别来自复合 KEM 的不同分量。对照查询是对已被给予这两份密钥份额的攻击者提出同样的查询，此时它们立即不成立。

- Result: 成立；回答了所问：是；可在有保留的情况下使用
- Value 中：这是该构造的核心机密性，除理想化原语外没有其他简化即得以证明。
- Stated in the documents: 是。文稿中所述位置：main:1210
- Rests on: SHA3-256、AES 密钥封装、ML-KEM 与 AEAD 都是理想的构造器和解构器，因此不捕获任何计算弱点。；Diffie-Hellman 只带有交换律；模型中不存在群阶或小阶点。；接收者的公钥被假定为真实可信，仅被直接发布；证书不在范围内。；数据包的长度、编码和八位组布局未建模。
- Open: 模型把接收者的公钥交给攻击者，但不给证书，因此这里没有回答任何关于密钥真实性的问题。
- Next step: 包含证书及其绑定签名的模型可以解决真实性部分。

K02: 消息明文的机密性

消息明文只对预期接收者可见，网络攻击者不能得到明文。

明文对网络上的攻击者保持隐藏：它用消息密钥加密，而消息密钥派生自攻击者无法获得的会话密钥。只有同时持有接收者两个私钥的人才能读取明文，对照查询通过把这些密钥交给攻击者来表明这一点。

- Result: 成立；回答了所问：是；可在有保留的情况下使用
- Value 中：这是用对称原语的常规理想化证明的一项核心保证。
- Stated in the documents: 是。文稿中所述位置：rfc9580:434
- Rests on: AEAD 是理想的认证加密：错误的密钥或 nonce 完全得不到明文。；从会话密钥和盐派生消息密钥是一个私有函数。；建模了一个分块。
- Open: 消息长度和分块数量未建模，因此不覆盖长度泄露。
- Next step: 要做流量分析，需要一个带填充和可观测长度的模型，而这是无法向证明器提出的。

A03: 关于所接受KEK的一致性

接收者成功解出会话密钥时，该 KEK 由发送者针对接收者公钥、算法 ID 与本次封装密文计算得到。

当接收者解封会话密钥时，它持有的 KEK 就是发送者针对其自身公钥、它所配置的算法 ID 以及它恰好收到的两个密文所计算的 KEK。单射解读不成立，因为同一条消息可能被投递给多个接收者运行，而且任何人都可以充当发送者。

- Result: 成立；回答了所问：是；可在有保留的情况下使用
- Value 低：它直接来自双方重新计算同一函数，因此只是确认了设计，并未发现新东西。
- Stated in the documents: 否

- Rests on: 该主张是针对发送者自身的封装表述的；没有发送者认证，攻击者构造的消息不构成违反。；KEK 被理想化为其输入的单射函数。；对照查询是单射变体，它不成立：一个发送者的封装可以被两个接收者运行接受。
- Open: 模型中只有一个接收者密钥对，因此无法表达针对第二个接收者的移植。
- Next step: 增加一个带有自己标识符的第二个接收者密钥对，就可以直接提出移植问题。

A08: KEK绑定到算法与密钥

接收者接受的 KEK 所绑定的 algId、ecdhPublicKey、ecdhCipherText 与其自身密钥和收到的 PKESK 一致，攻击者不能把为其他算法或公钥生成的封装挪用到本接收者。

KEK 绑定到算法 ID 和接收者的 ECDH 公钥：接收者绝不会接受在另一个复合算法下计算的 KEK，也绝不会接受为另一方的 ECDH 密钥计算的 KEK，即使 ML-KEM 密钥和 ECDH 密文相同也不行。

- Result: 成立；回答了所问：是；可在有保留的情况下使用
- Value 中：文稿把这一绑定作为覆盖 X25519 多目标攻击的理由；模型机械地再现了它。
- Stated in the documents: 是。文稿中所述位置：main:1214, main:1233
- Rests on: 假定 SHA3-256 的抗碰撞性；KEK 是其五个输入的理想函数。；第二个公钥是一个没有对应私钥的常量，因此该测试是移植测试，而不是完整的第二方。；每个主张的对照查询是一个可达性查询：它表明另一算法下的发送者、以及发往另一公钥的发送者确实会运行，因此接受之缺失并非空泛。没有查询能表明接收者接受了错误的 KEK，因为失败恰恰在于它从不接受。
- Open: 只建模了复合表中的两个算法 ID；无法替换其他套件。
- Next step: 对于这两种复合方案无需再做其他。

C08: One broken component keeps the KEK

在 X25519 分量失效（攻击者得到 ecdhKeyShare）或 ML-KEM 分量失效（攻击者得到 mlkemKeyShare）之一的情况下，KEK 与会话密钥仍对攻击者保密。

攻破复合 KEM 的一个分量不会泄露 KEK 或会话密钥：拥有 X25519 密钥份额的攻击者仍然需要该消息的 ML-KEM 密钥份额，反之亦然。同时拥有两份份额时 KEK 立即被推导出来，这正是对照查询。

- Result: 成立；回答了所问：是；可在有保留的情况下使用
- Value 中：这正是混合组合器存在的目的；证明成立，但附带原语被理想化这一保留。
- Stated in the documents: 是。文稿中所述位置：main:1210
- Rests on: 每个分量都被理想化为不可攻破的封装，只是泄露了被攻破的那一份份额。；攻破应用于一条已记录消息的密钥份额，这正是持有相应私钥所得到的结果。
- Open: ML-KEM 和 X25519 自身的强度未建模；只有组合的结构被建模。
- Next step: 带有真实代数和原语假设的 Tamarin+ Agent 可以解决计算方面的问题。

C01: 过去消息无前向保密

会话完成后泄露接收者长期私钥，攻击者仍不能得到此前会话的 KEK、会话密钥与明文。

一旦接收者的两个长期私钥都泄露，它已经收到的每条消息都可被读取：攻击者从 ECDH 私钥重新计算 X25519 密钥份额，用解封装密钥解封装 ML-KEM 密文，重建 KEK 并解封会话密钥。不存在前向保密，这源于加密到静态密钥。

- 结果：不成立（1 个攻击）；回答了所问：是；可用但有保留
- 价值低：这是 KEM 加密针对静态接收方密钥的固有性质，OpenPGP 的任何读者都不会期待其他结果。

- 文稿中是否陈述：不明确。所依据的原文：rfc9580:6277
- 依据：泄露会同时给出两把私钥；单把私钥不够，C08 已表明这一点。；攻击者保留录下的数据包，网络攻击者正是这样做的。
- 待定：泄露的私钥应仍保护什么，文稿未作定论，因此没有回答更强的解读。
- 下一步：要询问前向保密，需要交互式或棘轮式变体。

S01: AEAD nonce不复用

同一消息密钥下，AEAD nonce（IV 与块索引组合）不重复使用。

在同一条消息内，写入每个块的 nonce 在消息密钥下从不重复：消息密钥和 IV 来自新的 salt，块索引随每个块递增。有界模型在三个块上展示了这一点，并表明一旦索引不递增，该性质就不成立。

- 结果：成立；回答了所问：部分；可用但有保留
- 价值低：它确认了文稿规定的构造，并指出了它唯一依赖的东西：递增的索引。
- 文稿中是否陈述：是。文稿中的位置：rfc9580:6733
- 依据：三个块，而非任意数量；计数器被展开。；salt 被假设为新鲜，模型用一个新名称表示这一点；生成器的质量未被建模。；这个模型上的对照只表明块确实被封装；使该性质不成立的对照，即发送方不递增索引，在有界模型中。；64 位索引的回绕在状态审查中记录为未经确认的候选。
- 待定：计数器只展开到三个块，因此除了回绕观察之外，对非常长的消息没有结论。
- 下一步：证明器之外的计数论证可以判定 64 位索引是否会被耗尽。

K01.8：会话密钥与明文保密

攻击者不能得到会话密钥与消息明文。

只掌握网络的攻击者既得不到会话密钥，也得不到明文。一旦接收方的两把私钥泄露，两者都会丢失，这是该主张的边界，也是对照所表明的内容。

- 结果：成立；回答了所问：是；可用但有保留
- 价值中：它直接回答了用户关于保密性的问题，并附有标准保留。
- 文稿中是否陈述：是。文稿中的位置：rfc9580:434, main:1210
- 依据：原语是理想函数和析构器。；接收方的长期密钥被假设在会话期间留在接收方处。
- 待定：与 K01 相同的限制：原语理想化，无长度，无证书。
- 下一步：在此抽象层次上，该问题无需进一步工作。

A08.9：KEK绑定到algId、密钥、密文

接收者接受的 KEK 所绑定的 algId、ecdhPublicKey、ecdhCipherText 与其自身密钥和收到的 PKESK 一致。

被接受的 KEK 绑定到算法 ID、接收方的 ECDH 公钥以及它收到的两个密文：接收方针对发送方封装的 KEK 就是发送方的 KEK，在其他算法下或针对另一 ECDH 密钥所做的封装均不被接受。

- 结果：成立；回答了所问：是；可用但有保留
- 价值中：它用证明而非论证回答了用户的绑定问题。
- 文稿中是否陈述：是。文稿中的位置：main:1214, main:1233
- 依据：KEK 是其五个输入的理想单射函数。；另一个算法 ID 使用相同的密钥材料进行测试，这比完整的第二套方案更严格。；对照是可达性对照，表明另一个发送方可以运行；此外还有 KEK 约定的单射变体，该变体不成立。

- 待定：只存在两个算法 ID 和一个接收方密钥对，因此绑定到第三套方案或子密钥的问题未被回答。
- 下一步：加入第三个算法 ID 和一个子密钥可扩展绑定测试。

C08.10：组件泄露韧性并含双泄露对照

在 X25519 分量泄露、ML-KEM 分量泄露或两者同时泄露的情况下，检查 KEK 与会话密钥是否仍保密；单分量泄露应保持，双分量同时泄露作为对照。

仅攻破 X25519 分量不会泄露 KEK 和会话密钥，因为仍需要 ML-KEM 份额；仅攻破 ML-KEM 分量也不会泄露它们，因为仍需要 ECDH 份额。两者同时攻破时 KEK 立即失效，这就是表明该性质可能不成立的对照。

- 结果：成立；回答了所问：是；可用但有保留
- 价值中：它回答了用户的第三个问题，包括用户要求的双泄露对照。
- 文稿中是否陈述：是。文稿中的位置：main:1210
- 依据：除泄露的份额外，每个分量都被理想化。；对照是在同一模型上被证伪的查询，而非单独的协议。
- 待定：证明中每次只攻破一个分量；双攻破是对照，不是单独的目标。
- 下一步：在此抽象层次上无需进一步工作。

X01: SEIPD明文完整性

接收方从不释放在该会话下并非来自发送方的明文：如果它从 v2 SEIPD 数据包恢复出明文 m，则发送方在同一会话下封装了 m，因此任何被修改或伪造的密文都不会被当作新明文接受。

接收方不释放发送方未在该会话下封装的任何明文：被修改的密文、错误的密钥或错误的 nonce 完全不产生明文。任何人都可以用接收方的公钥加密，这不是完整性的破坏，而是不存在发送方认证的结果。

- 结果：成立；回答了所问：是；可用但有保留
- 价值中：拒绝畸形或被修改的密文正是 RFC 9580 13.7 要求的性质，模型重现了它。
- 文稿中是否陈述：是。文稿中的位置：rfc9580:6516, rfc9580:6520
- 依据：AEAD 是理想的认证加密，不存在伪造。；该主张是针对发送方自身的封装陈述的，因为攻击者构造的消息是合法的新消息。；一个块，无最终长度标签，无流式处理。；对照是一个可达性查询，表明接收方确实接受被封装的明文，而不是一个它接受错误明文的查询。
- 待定：最后若干块的截断和流式释放规则未被建模。；私钥泄露让攻击者可以构造自己的消息，该主张并不禁止这一点。
- 下一步：带最终长度标签的多块模型将覆盖截断。

X02: 逐消息KEK分离

两条消息从不共享 KEK：如果两个封装产生相同的 KEK，它们就是同一个封装，因此一条消息的 KEK 泄露不会透露另一条消息的任何信息。

两条消息从不共享 KEK：如果两个封装产生相同的 KEK，它们就是同一个封装。因此一条消息的 KEK 泄露不会透露另一条消息的任何信息。

- 结果：成立；回答了所问：是；可用但有保留
- 价值低：它来自每条消息的新鲜随机性；值得陈述，因为它限制了 KEK 泄露的损害。
- 文稿中是否陈述：否。所依据的原文：main:1219

- 依据：每条消息的新鲜密钥份额用新名称表示。；只比较两条消息；论证不计数会话。；对照只表明 KEK 确实被产生；该模型没有变体使两条消息共享一个 KEK。
- 待定：分离只对 KEK 展示，未对会话密钥重用回复中的消息密钥展示。
- 下一步：会话密钥重用特性的模型将把它扩展到同一会话密钥下的消息密钥。

附录 C：主张和证明器的答案

证明器对每个目标的记录：达到的级别、达到该级别的查询、是否检查了对照，以及每个攻击及分析者的判定。

目标	级别	界	查询	对照	攻击	主张
K01	proved		k01_kek	checked		4 proved
K02	proved		k02	checked		1 proved
A03	proved		a03	checked		1 proved
A08	proved		a08_alg	checked		2 proved
C08	proved		c08_pq	checked		2 proved
C01	none				c01 (holds_in_source)	1 attack
S01	proved		s01	checked		1 proved, 1 bounded
K01.8	proved		k01_sk	checked		2 proved
A08.9	proved		a08_alg	checked		2 proved
C08.10	proved		c08_pq	checked		2 proved
X01	proved		x01	checked		1 proved
X02	proved		x02	checked		1 proved

逐一按主张，采用分析者对每个主张的措辞；每个的答案是证明器的答案：

目标	主张	攻击者	回答	查询	对照
K01	网络上的攻击者无法从消息的 PKESK 和接收者公钥推导出该消息的 KEK。	network	proved	k01_kek	checked
K01	网络上的攻击者无法推导出发送方在 KEK 下包装的会话密钥。	network	proved	k01_sk	checked
K01	网络上的攻击者无法推导出接收者的 ECDH 私钥。	network	proved	k01_priv	checked
K01	网络上的攻击者无法推导出接收者的 ML-KEM 解封装密钥。	network	proved	k01_mlk	checked
K02	网络上的攻击者无法从 PKESK 和 SEIPD 包得到消息明文。	network	proved	k02	checked
A03	接收者针对发送方封装（相同 ecdhPublicKey、ecdhCipherText 和 mlkemCipherText）所接受的 KEK，就是发送方计算出的 KEK。	network	proved	a03	checked
A08	该接收者不接受在其他算法 ID（ML-KEM-1024 + X448，algId 36）下计算的任何 KEK。	network	proved	a08_alg	checked
A08	即使 ML-KEM 密钥和密文相同，该接收者也不接受为另一方 ECDH 公钥计算的任何 KEK。	network	proved	a08_key	checked

目标	主张	攻击者	回答	查询	对照
C08	在 X25519 分量被攻破（攻击者持有所记录消息的 ECDH 密钥份额）时，KEK 和会话密钥仍然保密：只有该消息的 ML-KEM 密钥份额也被泄露，它们才会泄露。	`ecdh_component_broken`	proved	c08_pq	checked
C08	在 ML-KEM 分量被攻破（攻击者持有所记录消息的 ML-KEM 密钥份额）时，KEK 和会话密钥仍然保密：只有该消息的 ECDH 密钥份额也被泄露，它们才会泄露。	`mlkem_component_broken`	proved	c08_cl	checked
C01	对于已经收到的消息，攻击者在运行结束后获得接收者长期私钥时，会话密钥和明文仍然对它隐藏。此性质不成立：两个密钥份额都是接收者静态密钥的函数，因此攻击者重新计算 X = X25519(r, V) 和 mlkemKeyShare = ML-KEM.Decaps(mlkemCipherText, sk)，然后得到 KEK，解包 C 并推导出消息密钥。	`ltk_leak_after_run`	attack (holds_in_source)	c01	
S01	同一消息密钥绝不会与同一个 AEAD nonce 相遇两次：在一条消息展开的三个分块中，nonce（IV ‖ 分块索引）只有在分块索引不前进时才会重复。	`network`	bounded（变化值展开为 3 步）	s01_chunks	checked
S01	同一消息密钥绝不会与同一个 AEAD nonce 一起使用两次：共享同一消息密钥和同一 nonce 的两次 v2 SEIPD 加密就是同一次加密。	`network`	proved	s01	checked
K01.8	网络上的攻击者无法获得消息的会话密钥。	`network`	proved	k01_sk	checked
K01.8	网络上的攻击者无法获得消息明文。	`network`	proved	k02	checked
A08.9	PKESK 中的算法 ID 被绑定进 KEK：只有用接收者自己的算法 ID 进行的封装才会得到被接受的 KEK。	`network`	proved	a08_alg	checked
A08.9	被接受的 KEK 绑定到接收者自己的 ECDH 公钥以及收到的 ecdhCipherText 和 mlkemCipherText：它正是发送方针对这些值计算出的 KEK。	`network`	proved	a03	checked
C08.10	仅攻破 X25519 分量并不能得到 KEK：攻击者还需要该消息的 ML-KEM 密钥份额。	`ecdh_component_broken`	proved	c08_pq	checked
C08.10	仅攻破 ML-KEM 分量并不能得到 KEK：攻击者还需要该消息的 ECDH 密钥份额。两个分量都被攻破时 KEK 被推导出来，这正是该性质可以失败所对应的对照。	`mlkem_component_broken`	proved	c08_cl	checked
X01	接收者只释放发送方为该会话封装好的明文：被修改或伪造的 SEIPD 密文不会被当作新的明文接受。	`network`	proved	x01	checked
X02	两条消息绝不会共享同一个 KEK：使用同一 KEK 的两次封装就是同一次封装。	`network`	proved	x02	checked

附录 D：分析是怎么跑的

一名 LLM 分析者根据文稿构建协议模型，并把每个性质交给 ProVerif；第 3 节的每个结果都是证明器给出的回答。

ProVerif+ Agent 会话 session-20261114T212745，2026-11-14。状态：已完成。模型：deepseek-v4-1-flash-261017。步数：61，证明器 6 s，实际耗时 14 min，成本 0.134 USD。

ProVerif：运行 106 次证明，105 次得出结论，1 次无答案，4 s，记录 30 个回答。

分析了什么

协议	OpenPGP composite KEM encrypted message
版本	rfc9980
分析范围	固定 ML-KEM-768+X25519 复合 KEM 的 PKESK 与 v2 SEIPD 单接收者单消息加密/解密；可信接收者公钥预先固定。不是交互握手，不凭加密声称发送者认证；排除签名、密钥分发、压缩及多接收者。
不在范围内	符号模型不证明算法的计算安全或抗量子安全；攻击轨迹只在明确假设下成立，不直接等同规范漏洞。每个性质必须有可达性/正例控制。
原文	251 个编号原文单元（40193 字节），来自下面列出的章节。本报告中诸如 s0001 的引用就是这些原文单元之一

分析所依据的文稿章节：

文稿	章节	标题
main	3	Algorithm Combinations
main	3.1	Composite KEMs
main	3.2	Composite Signatures
main	3.3	Multiple Signatures
main	3.4	ECC Requirements
main	3.5	Key Version Binding
main	4	Composite KEM Schemes
main	4.1	Building Blocks
main	4.1.1	ECDH KEM
main	4.1.1.1	X25519-KEM
main	4.1.1.2	X448-KEM
main	4.1.2	ML-KEM
main	4.2	Composite Encryption Schemes with ML-KEM
main	4.2.1	Key Combiner
main	4.2.2	Key Generation Procedure
main	4.2.3	Encryption Procedure
main	4.2.4	Decryption Procedure

文稿	章节	标题
main	4.3	Packet Specifications
main	4.3.1	Public Key Encrypted Session Key Packets (Packet Type ID 1)
main	4.3.2	Key Material Packets
main	4.3.2.1	Public Key Packets (Packet Type IDs 6 and 14)
main	4.3.2.2	Secret Key Packets (Packet Type IDs 5 and 7)
main	7	Notes on Algorithms
main	7.1	Symmetric Algorithms for SEIPD Packets
main	7.2	Hash Algorithms for Key Binding Signatures
rfc9580	5.1	Public Key Encrypted Session Key Packet (Type ID 1)
rfc9580	5.1.1	Version 3 Public Key Encrypted Session Key Packet Format
rfc9580	5.1.2	Version 6 Public Key Encrypted Session Key Packet Format
rfc9580	5.1.3	Algorithm-Specific Fields for RSA Encryption
rfc9580	5.1.4	Algorithm-Specific Fields for Elgamal Encryption
rfc9580	5.1.5	Algorithm-Specific Fields for ECDH Encryption
rfc9580	5.1.6	Algorithm-Specific Fields for X25519 Encryption
rfc9580	5.1.7	Algorithm-Specific Fields for X448 Encryption
rfc9580	5.1.8	Notes on PKESK
rfc9580	5.13	Symmetrically Encrypted and Integrity Protected Data Packet (Type
rfc9580	5.13.1	Version 1 Symmetrically Encrypted and Integrity Protected Data
rfc9580	5.13.2	Version 2 Symmetrically Encrypted and Integrity Protected Data
rfc9580	5.13.3	EAX Mode
rfc9580	5.13.4	OCB Mode
rfc9580	5.13.5	GCM Mode

源文步骤在文稿中定位到的协议用途：固定 ML-KEM-768+X25519 复合 KEM 的 PKESK 与 v2 SEIPD 单接收者单消息加密/解密；可信接收者公钥预先固定。不是交互握手，不凭加密声称发送者认证；排除签名、密钥分发、压缩及多接收者。

对照文稿定性质

在编写查询之前，分析者把草拟的性质清单与文稿本身对协议用途的说明逐条对照。性质的名称、性质所涉及的内容以及文稿的表述都是分析者的解读；所提及的位置确实存在于文稿中。

ID	简称	对象	文稿怎么说	出处
K01	KEK与私钥的保密性	发送方的 KEK、会话密钥，以及接收方的 ECDH 和 ML-KEM 私钥，这些都不让网络上的攻击者拿到。	第 4.2 节把两份密钥份额送入密钥组合器，产生唯一的共享秘密，用它来包裹会话密钥（s0066、s0067、s0072）；第 9.2 节论证该组合器具有 IND-CCA2 安全性，即 KEK 不可恢复；ECHD 和 ML-KEM 私钥假定由接收方持有。	main:1210, s0003, s0066, s0067, s0072, s0087

ID	简称	对象	文稿怎么说	出处
K02	消息明文的机密性	单条消息的明文，不让网络上的攻击者拿到，该攻击者可能持有每一个数据包。	会话密钥包裹内容加密；v2 SEIPD 以 AEAD 构造加密数据，所用的消息密钥由会话密钥和盐派生（s0204、s0231、s0233）。	rfc9580:434, s0004, s0067, s0202, s0204, s0231
A03	关于所接受KEK的一致性	接收方：当它解包会话密钥时，它接受的 KEK 是由发送方针对接收方自己的公钥、该算法 ID 和那些封装密文计算出来的。	解密流程根据收到的密文和接收方自己的公钥重新计算组合器（s0093、s0096、s0099），若解包完整性检查失败则中止（s0100）；文稿陈述了该构造，但没有把这一点本身作为目标陈述。	s0080, s0082, s0086, s0093, s0099, s0100
A08	KEK绑定到算法与密钥	针对某个接收方公钥和算法 ID 所接受的 KEK：为另一算法、另一 ECDH 公钥或另一 ECDH 密文所做的封装都无法得到它。	第 9.2.1 节指出，算法 ID 把派生的 KEK 绑定到所选算法（main:1233），并且把 ECC 公钥纳入组合器可以应对多目标攻击（main:1214）；RFC 7748 警告说，公钥若未被绑定进密钥派生，可能导致漏洞（rfc7748:820）。	main:1214, main:1233, rfc7748:820, s0066, s0072, s0093
C08	One broken component keeps the KEK	一条消息的 KEK 和会话密钥，不让攻破复合 KEM 两个组件之一（X25519 组件或 ML-KEM 组件）但未攻破另一组件的攻击者拿到。	第 9.2 节：若 ML-KEM 具有 IND-CCA2 安全性或 Strong Diffie-Hellman 问题成立，则组合器具有 IND-CCA2 安全性，因此一个组件的失效不足以攻破（main:1210）。	main:1210, s0003, s0070, s0072
C01	过去消息无前向保密	已接收消息的会话密钥和明文，不让在运行结束后获得接收方长期私钥的攻击者拿到。	文稿没有提及前向保密；它们只假定私钥部分由其所属方控制和保护（rfc9580:6277）。目标针对该密钥提出了反事实问题，而构造只使用接收方的静态 ECDH 密钥和静态 ML-KEM 解封装密钥作为两份密钥份额的唯一输入（s0064、s0084、s0085、s0097、s0098）。	rfc9580:6277, s0032, s0064, s0084, s0085, s0097, s0098
S01	AEAD nonce不复用	v2 SEIPD 数据包的 AEAD nonce，在同一消息密钥下绝不使用两次。	32 字节的盐必须安全生成（s0230）；消息密钥和 IV 都由 HKDF 从会话密钥和盐派生，nonce 就是该 IV 后接块索引（s0233、s0242）；RFC 9580 把盐和 nonce 列为 OpenPGP 公开可见的随机字段（rfc9580:6733）。	rfc9580:6733, s0230, s0233, s0238, s0242
K01.8	会话密钥与明文保密	一条消息的会话密钥和消息明文，不让网络上的攻击者拿到。	会话密钥按 RFC 9580 的规定生成，并在 KEK 下包裹（s0067）；接收方只有在密钥包裹完整性检查通过后才输出它（s0100、s0101）。	main:1210, rfc9580:434, s0067, s0100, s0202
A08.9	KEK绑定到algId、密钥、密文	针对某个接收方公钥和算法 ID 所接受的 KEK，以及它所绑定的 ECDH 密文。	第 9.2.1 节指出，算法 ID 把 KEK 绑定到算法，并且 ECC 公钥是组合器的输入（main:1214、main:1233）。	main:1214, main:1233, s0066, s0072, s0093

ID	简称	对象	文稿怎么说	出处
C08.10	组件泄露韧性并含双泄露对照	KEK 和会话密钥，不让只攻破 X25519 组件或只攻破 ML-KEM 组件的攻击者拿到，并以两个组件都被攻破作为对照。	第 9.2 节：若 ML-KEM 具有 IND-CCA2 安全性或 Strong DH 成立，则组合器的 IND-CCA2 安全性成立，即一个组件被攻破仍可存活（main:1210）。	main:1210, s0003, s0070, s0072
X01	SEIPD明文完整性	接收方和 v2 SEIPD 数据包的明文；这是消息完整性，不是发送方认证。	RFC 9580 13.7 要求，AEAD 标签校验失败时实现 MUST NOT 解析或释放已解密数据，因为在确认完整性之前释放可能泄露数据（rfc9580:6516、rfc9580:6520），并把块标签校验失败的 v2 SEIPD 数据包列为可延展密文（rfc9580:6548）；JKS02 解密预言机攻击被作为密文可延展性的一种形式提及（rfc9580:6330）。	rfc9580:6516, rfc9580:6520, rfc9580:6548, rfc9580:6330, s0204, s0231, s0238
X02	逐消息KEK分离	单条消息的 KEK，与所有其他消息的 KEK 保持分离。	第 9.2.1 节给出 domSep 及其长度字节，使密钥材料只用于一个目的（main:1219），并且两份密钥份额都是每条消息新生成的（s0084、s0085）。	main:1219, s0072, s0084, s0085

分析者依据文稿补充的性质：X01, X02.

文稿里的安全表述，逐段去向

宿主把文稿里说明协议保护什么、防什么的章节逐段列出，共 110 段；分析者对每一段说明了去向：定为性质、定为攻击者条件，或者为什么都不选。选了 17 段，没选 93 段。列哪些段落由标准的章节结构决定；每一段的去向是分析者的判断。

段落	出处	定为	不选的原因
P001 至 P002	main:1163-1187	不在范围内	复合签名的签名剥离与 SUF-CMA；任务排除了签名。
P003	main:1191-1204	不在范围内	签名的跨协议重用；任务排除了签名。
P004	main:1208-1212	C08, C08.10	
P005	main:1214-1215	A08	
P006	main:1219-1231	X02	
P007	main:1233-1237	A08, A08.9	
P008 至 P009	main:1241-1257	不在范围内	Hedged ML-DSA 与 SLH-DSA，以及签名的摘要长度。
P010 至 P011	main:1261-1272	证明器问不了	该优先选哪种对称算法属于原语的强度，而不是迹性质。
P012	main:1276-1281	K01, `ecdh_component_broken`, `mlkem_component_broken`	
P013	main:1285-1287	证明器问不了	随机生成器的质量无法建模；模型把新鲜名视为不可猜测。
P014 至 P016	rfc7748:783-818	证明器问不了	curve25519/curve448 的具体安全强度，以及 X25519 的小阶输入或非贡献输入：符号模型没有群阶，也没有余因子，因此无法表示这些内容。

段落	出处	定为	不选的原因
P017	rfc7748:820-825	A08, A08.9	
P018	rfc7748:827-837	证明器问不了	曲线实现之间可观察的差异（扭点、非规范值）属于实现行为。
P019	rfc9580:6268-6274	没有陈述目标	建议关注文献中已被攻破的算法。
P020	rfc9580:6276-6278	K01	
P021	rfc9580:6280-6282	证明器问不了	MD5 与 SHA-1 的弱点属于哈希原语的强度。
P022	rfc9580:6284-6289	不在范围内	将加密密钥与签名密钥分开；任务排除了签名。
P023	rfc9580:6291-6295	不在范围内	DSA 签名哈希替换。
P024	rfc9580:6297-6304	证明器问不了	消息中最弱元素足够强，这是关于原语强度的陈述。
P025 至 P027	rfc9580:6306-6322	不在范围内	哈希碰撞、签名中替换哈希算法，以及首选签名算法。
P028	rfc9580:6324-6328	证明器问不了	对称算法被分析得有多充分，无法向证明器提问。
P029	rfc9580:6330-6339	X01	
P030 至 P032	rfc9580:6343-6361	不在范围内	签名中的 SHA-1 碰撞检测。
P033 至 P035	rfc9580:6365-6390	不在范围内	加盐签名及其优点。
P036	rfc9580:6394-6404	证明器问不了	椭圆曲线侧信道属于实现性质。
P037 至 P040	rfc9580:6408-6431	不在范围内	v1 SEIPD/SED 快速检查与 v4 SKESK 口令；所建模的流程是带 v2 SEIPD 包的 v6 PKESK，其中 s0147 与 s0155 排除了该检查。
P041	rfc9580:6433-6435	X01	
P042 至 P044	rfc9580:6439-6461	不在范围内	RSA 与 ElGamal 加密的 PKESK 中的 PKCS#1 填充错误；这些算法不在复合 KEM 范围内。
P045 至 P051	rfc9580:6465-6498	没有陈述目标	指纹对人类可用性。
P052 至 P054	rfc9580:6502-6521	X01	
P055	rfc9580:6523-6526	不在范围内	从历史非 AEAD 包中释放明文；所建模的流程仅限 AEAD。

段落	出处	定为	不选的原因
P056	rfc9580:6528-6533	不在范围内	在截断尾部之前流式释放完全认证的分块；模型只有一个分块，没有流式处理。
P057	rfc9580:6535-6536	X01	
P058 至 P060	rfc9580:6538-6545	不在范围内	SED 包、压缩数据失败与 v1 SEIPD MDC 失败；这些都不在所建模的流程中。
P061	rfc9580:6547-6549	X01	
P062 至 P065	rfc9580:6551-6560	不在范围内	静态存储的加密私钥材料的完整性。
P066	rfc9580:6562-6565	没有陈述目标	关于选择哪种容器格式的建议。
P067	rfc9580:6567-6567	不在范围内	已弃用的 SED 包。
P068 至 P069	rfc9580:6569-6577	没有陈述目标	建议在接收方支持时使用 v2 SEIPD 包加密；它陈述的是格式选择，而不是本协议的性质。
P070	rfc9580:6579-6580	不在范围内	当多个接收方中有一个不支持 v2 时回退到 v1 SEIPD；任务排除了多个接收方。
P071	rfc9580:6582-6590	不在范围内	用于口令保护的私钥材料的 AEAD，以及 CFB 加密私钥的损坏。
P072 至 P073	rfc9580:6592-6595	没有陈述目标	面向实现者和用户的迁移建议。
P074	rfc9580:6599-6609	S01	
P075 至 P083	rfc9580:6611-6677	不在范围内	会话密钥重用回复功能：接收方子集、添加接收方、草稿与部署指导。任务建模的是一条消息发给一个接收方、使用新鲜会话密钥。
P084 至 P093	rfc9580:6681-6717	不在范围内	托管式撤销签名。
P094 至 P095	rfc9580:6721-6731	证明器问不了	对 CSPRNG 的要求以及随机数据可见性的级别；模型把每个新鲜名都视为不可猜测。
P096	rfc9580:6733-6734	S01	
P097	rfc9580:6736-6737	K01, K01.8	
P098 至 P100	rfc9580:6739-6750	证明器问不了	被攻破的 CSPRNG 会允许什么，以及按可见性级别分别使用生成器。
P101 至 P104	rfc9580:6754-6773	证明器问不了	通过消息长度进行流量分析；证明器没有长度概念，也没有两个密文大小相等的概念。

段落	出处	定为	不选的原因
P105	rfc9580:6777-6782	不在范围内	签名被偷偷转发。
P106 至 P108	rfc9580:6786-6808	不在范围内	经过哈希和未经哈希的签名子包。
P109 至 P110	rfc9580:6812-6820	不在范围内	恶意压缩数据；任务排除了压缩。

性质目录，逐条去向

性质是从一份安全和隐私性质目录起草的。起草时有 35 条没选，分析者把它们逐条过了一遍：作为性质在这里检查，或者写明原因不检查。检查了 7 条，没检查 28 条。“文稿没有提到”不能作为不检查的原因。

条目	目录里的性质	作为哪条性质检查	不检查的原因
K03	强保密／秘密值不可区分性	K01, K02	
K04	密钥用途、方向与会话分离	X02	
D01	消息完整性与数据来源真实性	X01	
A01	对端曾参与性（Aliveness）	不适用	单向消息加密：接收方解密一个存储的包，并不涉及对端是否运行过协议。
A02	弱一致性认证	不适用	任务声明加密不主张发送方认证，因此无法要求在对端身份上达成一致。
A04	单射一致性认证	不适用	同一密文可以被合法地再次解密；文稿不主张重放保护，也不主张只接受一次。
A05	隐式密钥认证	K01	
A06	显式密钥确认	不适用	没有密钥确认：发送方不等待回应。
A07	身份—凭据—密钥绑定	不在范围内	任务排除了证书、密钥分发以及 PKESK 中标识接收方的字段；接收方公钥是固定且可信的。
R01	认证／挑战新鲜性	不适用	没有挑战、时间戳、票据或新鲜性检查。
R02	消息级抗重放	不适用	不主张消息级重放保护；重新解密一条已存储的消息不算违反协议。
R03	业务操作至多一次	不适用	没有具备至多一次语义的计费或消耗资源的操作。
R04	消息接受顺序／窗口安全	不适用	单条消息，没有序号；AEAD nonce 的分块索引由 S01 覆盖。
R05	抗降级／安全策略保持	A08, A08.9	
C02	抗密钥泄露冒充（KCI）	不适用	单向加密没有对端认证，因此密钥泄露冒充不适用。
C03	已知密钥安全／泄露隔离	X02	
C04	临时秘密／部分状态泄露韧性	C08, C08.10	
C05	泄露后安全恢复（PCS）	不适用	没有棘轮或密钥更新机制，因此被攻破后无法恢复。
C06	注册凭据泄露后的增强认证	不适用	不是 PAKE；没有注册记录。
C07	抗离线字典攻击	不适用	不涉及口令或其他低熵秘密。
Z01	授权执行正确性	不适用	对受保护操作没有授权决策。
Z02	委托权限不扩张	不适用	没有委托或代理机制。

条目	目录里的性质	作为哪条性质检查	不检查的原因
Z03	授权与身份／会话／操作绑定	不适用	没有授权令牌或会话绑定。
Z04	撤销与过期生效	不适用	所建模的流程中没有撤销或过期。
Z05	跨租户／跨服务安全隔离	不适用	没有租户、服务或安全域。
S02	重传／密钥重安装安全	不适用	没有密钥更新或重传状态机。
S03	抗回滚与状态单调性	不适用	没有可回滚的版本或计数器。
S04	跨阶段安全上下文一致性	不适用	没有多阶段生命周期。
S05	终止与旧状态失效	不适用	没有会话拆除，也没有可能被恢复的旧状态。
P01	身份匿名性	不在范围内	s0200 中的匿名接收方 PKESK 是标识接收方字段的一种可选模式，任务将其略去；所建模的流程根本不携带接收方标识。
P02	会话不可关联性	不适用	流程中没有临时身份或按会话使用的假名。
P03	用户／凭据存在性隐藏	不适用	没有用户或凭据查找。
V01	条件性进展／错误恢复	证明器问不了	进展与错误恢复无法向证明器提问。
V02	无非预期死锁	证明器问不了	无死锁不是可以向证明器提问的迹性质。
V03	受限资源滥用／抗放大	证明器问不了	资源消耗与放大无法在事件之外计数。

参与方和受保护的值，逐项是否设为攻击者条件

宿主把起草的性质里点到的参与方和受保护的值逐项列出，共 21 项；分析者对每一项说明了哪个攻击者条件控制这个参与方或持有这个值，或者为什么不设。设了攻击者条件的 6 项，没设的 15 项。清单来自性质本身；每一项怎么处理是分析者的判断。

ID	参与方或值	类别	攻击者条件	不设的原因
S01	发送者	参与方	攻击者得不到更多	发送者只持有自己选择的明文和自己生成的会话密钥；没有任何目标对发送者的诚实性提出要求，接收者也没有得到关于消息发送者身份的任何保证。
S02	接收者	参与方	`ltk_leak_after_run`	
S03	攻击者	参与方	攻击者得不到更多	这是网络本身的攻击者，它已经存在；它不持有其他任何东西。
S04	KEK	值	按定义全部失守	按定义，持有 KEK 的人就能解包该消息的会话密钥；在该攻击者条件下，该消息已没有任何本协议的性质可供追问。
S05	会话密钥 sessionKey	值	按定义全部失守	持有会话密钥的人可以导出消息密钥并解密该消息。
S06	接收者 ECDH 私钥	值	`ecdh_component_broken`	
S07	接收者 ML-KEM 私钥	值	`mlkem_component_broken`	
S08	消息明文	值	按定义全部失守	持有明文本身就是明文机密性的丧失。
S09	ecdhPublicKey	值	攻击者得不到更多	接收者的 ECDH 公钥在消息之前就交给发送者；它是公开的。
S10	algId	值	攻击者得不到更多	算法 ID 以明文形式在 PKESK 中传输。
S11	ecdhCipherText	值	攻击者得不到更多	ECDH 密文在网络上传输。
S12	mlkemCipherText	值	攻击者得不到更多	ML-KEM 密文在网络上传输。
S13	会话密钥	值	按定义全部失守	与 S05 是同一个值：持有会话密钥的人可以解密该消息。

ID	参与方或值	类别	攻击者条件	不设的原因
S14	ecdhKeyShare	值	`ecdh_component_broken`	
S15	mlkemKeyShare	值	`mlkem_component_broken`	
S16	历史会话密钥	值	按定义全部失守	持有过去的会话密钥就是该消息机密性的丧失。
S17	历史明文	值	按定义全部失守	持有过去的明文本身就是明文机密性的丧失。
S18	接收者长期私钥	值	`ltk_leak_after_run`	
S19	消息密钥	值	按定义全部失守	持有消息密钥的人可以解密该消息。
S20	AEAD nonce	值	攻击者得不到更多	nonce 由 SEIPD 包中以明文传输的盐和分块索引导出。
S21	盐	值	攻击者得不到更多	32 字节的盐是 v2 SEIPD 包的一个字段，是公开的。

每条性质在各攻击者条件下的答案

这些攻击者由分析者设定，在编写查询之前就已确定；每个答案都是在其中一个攻击者条件下记录的。表格单元格是证明器在该攻击者条件下对该性质给出的答案；“无答案”表示该性质在该攻击者集合下，证明器未能对某项主张作出判定，或没有记录任何结果；空单元格表示分析者没有为该性质设定该攻击者。

攻击者	持有什么	依据	出处
`network`	仅网络：它可以读取、丢弃、修改和发送消息，不持有任何一方的秘密		
`ecdh_component_broken`	某条已记录消息的 ECDH 密钥份额，等价于接收者的 X25519 私钥 r：有了 r 和已记录的 ecdhCipherText V，攻击者可以计算 X = X25519(r, V)。ML-KEM 组件保持完好。	第 9.2 节要求组合器在其两个输入中只有一个受到保护时仍保持 IND-CCA2 安全（main:1210）；目标 C08，以及接收者 ECDH 私钥（S06）和 ecdhKeyShare（S14）的暴露，都是在该攻击者条件下作答的。	main:1210, s0003, s0072, s0037
`mlkem_component_broken`	某条已记录消息的 ML-KEM 密钥份额，等价于 ML-KEM 解封装密钥：有了它，攻击者可以对已记录的 mlkemCipherText 运行 ML-KEM.Decaps。X25519 组件保持完好。	同样来自“一个组件失效后仍须可存活”这一要求（main:1210）；接收者 ML-KEM 私钥（S07）和 mlkemKeyShare（S15）的暴露按该攻击者解读。	main:1210, s0003, s0053, s0098
`ltk_leak_after_run`	接收者的两把私钥，即 ECDH 秘密密钥 r 和 ML-KEM 解封装密钥，在消息被接收并解密之后获得。	文稿假定私钥部分由其所属方控制和保护（rfc9580:6277），并未提及之后的失陷；目标 C01 追问的正是这一反事实情形，接收者长期私钥（S18）和接收者参与方（S02）的暴露按该攻击者解读。	rfc9580:6277, s0094, s0097, s0098

ID	`network`	`ecdh_component_broken`	`mlkem_component_broken`	`ltk_leak_after_run`
K01	成立			
K02	成立			
A03	成立			
A08	成立			
C08		成立	成立	
C01				不成立
S01	成立			

ID	network	ecdh_component_broken	mlkem_component_broken	ltk_leak_after_run
K01.8	成立			
A08.9	成立			
C08.10		成立	成立	
X01	成立			
X02	成立			

原文和目标

来源：251 个原文单元，40193 字节，冻结于 source-delivery-01.json。 目标：7 个已批准（K01、K02、A03、A08、C08、C01、S01）；提出的问题：3 个。 目标列表已按草稿事先批准，批准时使用了 --approve-targets。

宿主按其文档所述默认值（任务 `decisions: auto`）作出的决定，供你核对：

- approve_package: 批准源文包 protocol-3afdbf21efd53284a5be01418a83

本步骤的自述：1 次请求（1 次失败），0.000 USD，输入 34295 tokens，输出 4078，耗时 27 秒。

关于证明器和模型的说明

- S01：在 ProVerif 中作答，分析者为此目标指定了 changing_state

一个简化了代数（xor、dh_algebra、other_algebra）的 ProVerif 模型给攻击者的能力弱于真实运算：在其中的证明是针对更弱攻击者的证明，在其中的攻击则是攻击。一个省略了对会变化的值的检查（changing_state）的 ProVerif 模型让参与方接受更多：在其中的证明成立，在其中的攻击可能只是缺少了该检查。Tamarin+ Agent 按原样对这些建模。

分析者的状态复查

这一部分是分析者自己对原文的解读，不是证明器的答案。ProVerif 的“由 ProVerif 显示”一栏为“是”，表示 ProVerif 在展开的模型中找到了包含这些步骤的轨迹，因此这些步骤是可能的；它们是否破坏了某个目标，只由答案表中的一次攻击来说明，它们还付出了什么代价，则属于分析者的解读。“未找到”只说明查看了哪些模式，本身不能证明任何东西。

值	由谁保持	何时变化	何时检查	出处
v2 SEIPD 消息的 64 位 AEAD 分块索引	发送者（执行加密的实现），在一条消息的持续时间内	同一条 v2 SEIPD 消息的每个分块递增一；第一个分块的索引为零	接收者根据该分块在流中的位置重建 nonce（IV 来自 HKDF，最右侧 64 位为索引），并由 AEAD 标签验证；最后一个标签覆盖明文的总字节数	s0242, s0238, s0236
32 字节的 SEIPD 盐	发送者，每条消息一份；接收者从包中读取	每条消息都重新生成；复用会话密钥的回复会使用不同的盐以获得新的消息密钥	没有任何人进行比较；它是导出消息密钥和 IV 的 HKDF 盐，因此不同的盐会得到不同的消息密钥	s0230, s0233, s0234

模式	结果	目标	证明器显示	步骤
replay	未发现任何问题			在较早分块的位置重放一个分块，得到的是该位置对应的 nonce，而不是它原始索引对应的 nonce，因此其标签校验失败；有界模型只接受发送方在索引 0 处封装的那个分块。

模式	结果	目标	证明器显示	步骤
repeat_one_time_action	未发现任何问题			分块索引是每条消息的本地计数器；不会安装任何密钥，重复的消息也无法让计数器回退。在三个分块的情形下，有界模型显示出三个不同的 nonce。
desynchronisation	未发现任何问题			索引不是共享状态：接收方根据分块在流中的位置推导出它，因此任何两方都不可能持有不同的索引值。
update_before_check	未发现任何问题			发送方在封装分块之后才推进索引，并且不做任何校验，因此不会有值在检查之前发生变化。
no_rollback_on_failure	未发现任何问题			中止的加密不会留下任何被接受的消息：接收方不接受标签校验失败的分块，并且 salt 对下一条消息是新鲜的。在后来的失败之前流式发出分块的情形不在模型范围内。
race_between_sessions	未发现任何问题			salt 和会话密钥对每条消息都是新鲜的，X02 表明两条消息从不共享 KEK，因此两个并发的加密既不共享消息密钥，也不共享索引。
distinguishable_failure	未发现任何问题			模型中没有失败应答：标签校验失败根本不产生任何输出，因此攻击者没有可比较的可区分回复。
wrap_or_window	候选		否：未用证明器尝试	发送方从零开始分块索引，并每个分块加一；s0242 将其固定为 64 位大端值，但没有规定索引耗尽时会发生什么。 / 一条超过 2^64 个分块的消息会让索引回到同一消息密钥下已经用过的值，因为消息密钥和 IV 是每条消息从会话密钥和 salt 一次性推导出来的（s0233）。 / 于是两个分块在同一消息密钥下共享一个 nonce；当索引未推进时，有界模型正好展示了这一失败。
other	未发现任何问题			丢弃末尾分块会被基于总八位组计数的最终标签检测到，而且源文允许在报错的同时释放已经认证过的分块（rfc9580:6531）；我的模型只有一个分块且没有最终标签，因此对截断不作任何说明。

分析者为目标命名的内容：

步骤	目标	特征	原因
52	K01, K02, A03, A08, C08, C01, K01.8, A08.9, C08.10, X01, X02	plain_primitives	这些答案只依赖于哈希（以 SHA3-256 作为组合器）、带密钥解包的 AEAD 式对称加密、KEM 解封装，以及作为普通密钥协商使用的 Diffie-Hellman（唯一代数律是 g^xy = g^yx，没有其他代数律）。因此模型给攻击者的不多不少，正是这些原语的标准符号化处理；SHA3-256、ML-KEM 和 AES-KeyWrap 的计算强度未被建模。
52	S01	changing_state	AEAD nonce 是 HKDF 推导出的 IV 后接 64 位分块索引，发送方从零开始并每个分块加一（s0242）；因此 nonce 是否在同一消息密钥下重复取决于这个变化的计数器。model.pv 只有一个分块，因此该目标的答案由 bounded.pv 补全，其中发送方展开为三个分块，并按照源文所述对索引进行比较和推进。

分析者的结束说明

RFC 9980 的 ML-KEM-768 + X25519 复合 KEM PKESK（v6）配 v2 SEIPD 消息的模型，单接收方、单条消息：12 个目标，11 个已证明，1 个攻击，全部配有对照查询（model.pv，外加 bounded.pv，用于唯一一个依赖变化的分块索引的目标）。

已确立：KEK、会话密钥、明文以及接收方的两个私钥都保持在网络攻击者触及范围之外（K01, K02, K01.8）；接收方的 KEK 是发送方针对其自身公钥、算法 ID 和所收到密文计算出的 KEK，并且无论在另一个复合算法 ID 下进行封装，还是针对另一方的 ECDH 密钥进行封装，都永远不会被接受（A03, A08,

A08.9）；攻破复合 KEM 的一个组件得不到任何东西，两个都攻破才能得到 KEK（C08, C08.10）；接收方不会释放任何并非为该会话封装过的明文（X01）；两条消息从不共享 KEK（X02）；在一条消息内，AEAD nonce 从不在一个消息密钥下重复，这一点在三个分块展开中成立，且仅在分块索引被推进时成立（S01，有界，对照查询 s01_bug）。

不成立：C01，前向保密。持有两个长期私钥的攻击者可以重新计算 X = X25519(r, V)、解封装 ML-KEM 密文、重建 KEK、解包会话密钥并推导出消息密钥，因此每条记录下来的消息都可读。这是加密到静态密钥的固有性质；文稿对前向保密只字未提，因此它被报告为低价值发现，新颖性不明确，而不是作为新消息。

阻碍与限制：SHA3-256、AES-KeyWrap、ML-KEM 和 AEAD 被理想化；Diffie-Hellman 只带有 g^xy = g^yx，因此 RFC 7748 关于小阶点和等价公钥的警告在这里无法测试（Tamarin+ Agent 可以解决它们）。长度、v3 PKESK 对称算法长度检查、证书、接收方匿名性、多接收方消息、截断和流式处理都不在模型范围内。状态审查没有在分块索引中发现可利用之处；其 64 位回绕被记录为一个未确认的候选，需要 2^70 个八位组。