CMS ML-KEM AuthEnvelopedData 的安全性分析

本报告的其他语言版本：English。

rfc9936 · ProVerif+ Agent 报告

文稿	Use of ML-KEM in the Cryptographic Message Syntax (CMS)（协议定义）; Using Key Encapsulation Mechanism (KEM) Algorithms in the Cryptographic Message Syntax (CMS)（辅助）; Cryptographic Message Syntax (CMS) Authenticated-Enveloped-Data Content Type（辅助）
检查的性质	15 条：9 条成立，6 条不成立（8 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

采用 ML-KEM 的 CMS AuthEnvelopedData 将一个加密消息从发起方传送给接收者。发起方持有接收者经过认证的 ML-KEM-768 公钥，选择内容和认证属性；接收者持有对应的私钥；攻击者掌控网络。

发起方选取一个新鲜的内容认证加密密钥，为接收者封装一个共享密钥，用 HKDF 基于算法标识从中派生出一个成对密钥加密密钥，在该密钥下包裹内容密钥，并以属性作为关联数据对内容进行认证和加密。接收者用自己的私钥解封装，派生出相同的密钥加密密钥，解包内容密钥，验证标签并释放内容。

交换结束后，接收者确信其所验证的内容和属性是由某个运行了本协议的人发送的，且途中未被篡改。但它并不确信那个人是谁；发起方也得不到任何回应，因为没有回复。接收者的私钥日后一旦泄露，此前已发送给它的每条消息都将失去保密性。

各方：

- **发起方**：编写并加密消息。持有：接收者经过认证的 ML-KEM-768 公钥 pk；内容 M；认证属性 A。
- **接收者**：打开信封。持有：ML-KEM-768 私钥 sk；带有 rid 的证书。

图 1 展示了文稿所规定的交换过程。

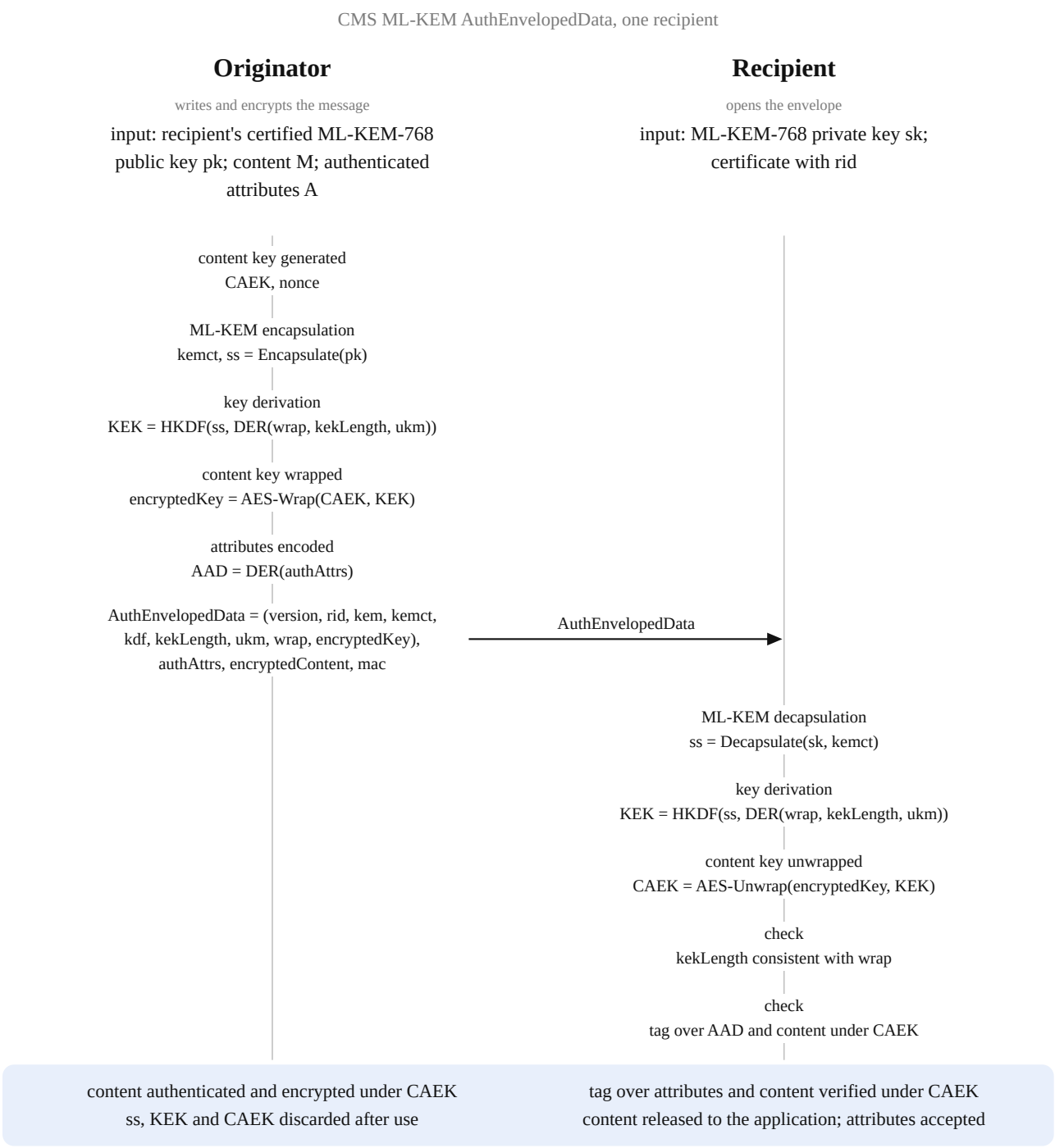


图 1. 文稿所规定的协议。

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。

该内容类型提供保密性、密钥保密性和标签保护的完整性，但不认证发起方；第二点在文稿中有说明。在检查的十五条性质中，九条成立、六条不成立：这些不成立分别是缺少数据源认证、私钥泄露导致历史消息失守、单条消息因其 KEK 泄露而失守，以及重放。十五条答案中有十四条是复现文稿已记录的结果；对

已被接受的 AuthEnvelopedData 的重放是文稿唯一没有写明的一点。所有答案都建立在 KEM、HKDF、密钥包裹和 AEAD 均视为理想的符号模型之上，因此它们对 ML-KEM-768 或 SHA-256 的计算强度不置一词。

攻击图沿用图 1 的各列。红色表示攻击者的行为。黄色标记的是一条消息，该消息由某一方发送，尽管规定的交换在该位置并没有这条消息，同时也标记攻击所破坏的结果。

2.1 一条被录制的加密消息被再次接受

类型	协议规范本身的弱点
是否新发现	是：文稿里没有写到
价值	低
涉及的性质	X04 （见第 3 节）
原文	s0105, s0106, s0112
依据	ProVerif attack query `replay`; Figure 2

发生了什么。 接收者不记录已经接受过的消息，KEMRecipientInfo 和 AuthEnvelopedData 也都不携带它所检查的计数器、时间戳或挑战值。因此，一条被录制下来的 AuthEnvelopedData 若被再次投递，会再次解密、再次验证，接收者会把同样的内容和同样的认证属性第二次当作新消息接受。

攻击者需要什么。 攻击者从网络上录制一条 AuthEnvelopedData，然后再次投递，一次或多次都可以。它不需要任何密钥，也不需要任何配合。

攻击者能得到什么。 接收者会对同一条消息执行两次动作：它把同一份明文释放两次；如果认证属性携带了某项指令，它会把该指令执行两次。重复投递还让攻击者能够使一条消息无限期地保持有效。

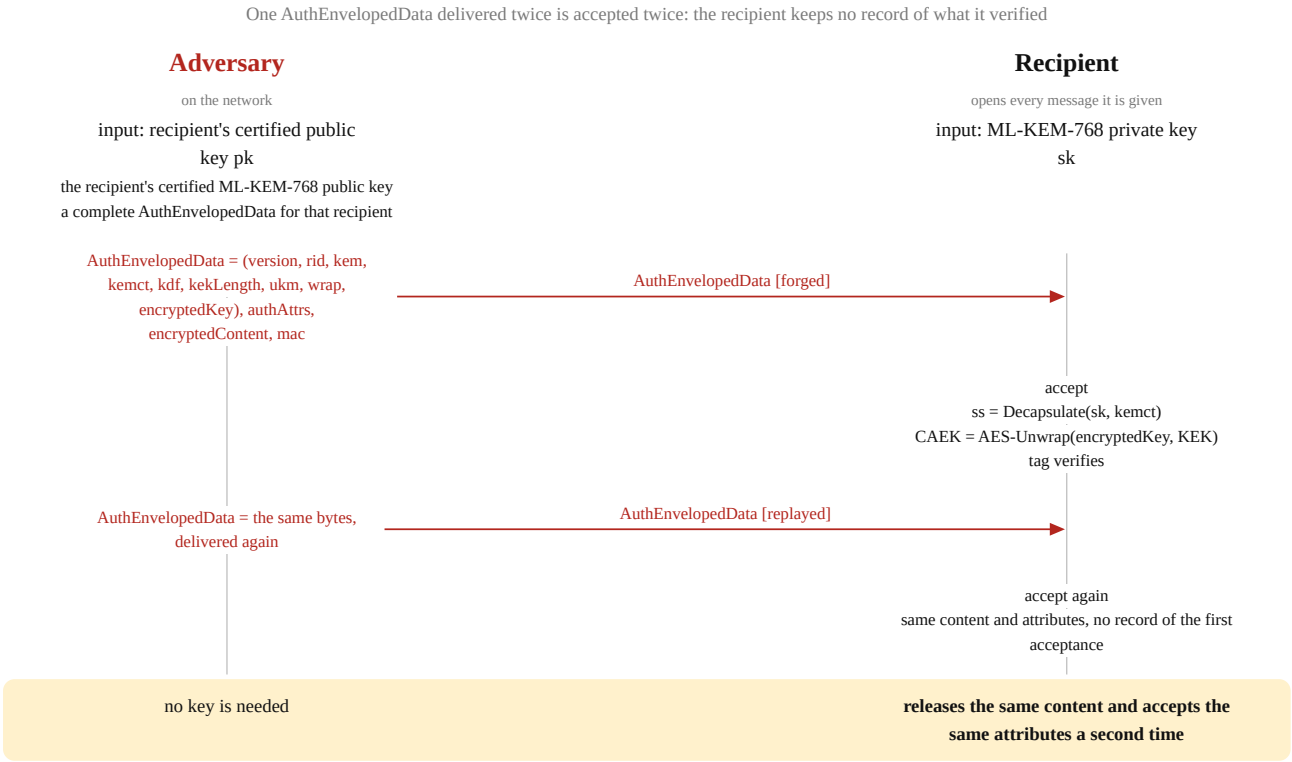


图 2. 一条 AuthEnvelopedData 被投递两次就被接受两次：接收者不记录自己验证过什么。性质 X04；攻击查询 replay。

拟议修改。 无。修改消息定义无法消除它：接收者必须记录自己已接受的内容，或者检查一个该内容类型并未定义的新鲜度值。ProVerif 的表只会增长，因此修复后的性质在本模型中无法对任意数量的会话证明，而且这一改动属于应用配置，而不属于 KEMRecipientInfo 或 AuthEnvelopedData 的定义。

3 检查过的性质

每一行是提交给 ProVerif 的一条性质。结果是 ProVerif 对分析者的协议模型给出的答案；价值和文稿是否已写明该结果，是分析者的判断，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	**私钥、KEK、CAEK 的保密性。** 攻击者在不知道接收者 ML-KEM 私钥、KEK、CEK/CAEK 的情况下，不能推出这些密钥中的任何一个。	无	成立	中	是
K02	**明文内容保密性。** 除发起方与目标接收者外，任何一方都不能从 AuthEnvelopedData 中推出被加密内容的明文。	无	成立	中	是
D01	**内容与属性的一致性。** 接收者接受的每条已验证内容与认证属性，都对应发起方真实发送的同一内容与属性，未被攻击者替换或修改。	无	攻击成立（文稿已写明）	中	是
A08	**KEMRecipientInfo 字段的绑定。** 接收者接受的 KEK 与 CEK 绑定到发起方实际选择的接收者标识、KEM 密文、KDF 上下文（wrap、kekLength、ukm）与算法标识，任一字段被替换都会导致验证失败。	无	成立	中	是
A05	**成对 KEK 与 CAEK 的隔离。** 对每个接收者，KEK 与 CEK 只对发起方和该接收者可得，其他实体不能推出。	无	成立	中	是
C01	**针对私钥泄露的前向保密。** 在会话完成且临时密钥擦除之后泄露接收者 ML-KEM 私钥，攻击者仍不能推出此前已完成会话的 CEK 与内容。	接收者 ML-KEM 私钥 (会话完成后)	攻击成立（文稿已写明）	中	是
C04	**临时组件泄露的后果。** 在指定临时组件（KEK、CEK/CAEK、封装随机值）之一泄露时，检查其余保密与完整性目标是否仍成立。	临时组件（KEK、CEK/CAEK、封装随机值）(会话进行中)	攻击成立（文稿已写明）	低	是
K02.8	**CEK 与明文保密性。** 攻击者不能推出 CEK 或被加密内容的明文。	无	成立	中	是
A08.9	**KEMRecipientInfo 字段绑定。** 接收者接受的 KEK 与 CEK 绑定到发起方实际选择的接收者标识、KEM 密文、KDF 上下文与算法标识。	无	成立	中	是

ID	性质	假定的泄露	结果	价值	文稿是否已写
D01.10	**内容与属性一致性。** 接收者接受的已验证内容与认证属性对应发起方真实发送的同一内容与属性，未被篡改。	无	攻击成立（文稿已写明）	中	是
X01	**每条消息使用新的 KEM 共享密钥。** KEM 共享密钥以及由它派生出的 KEK 只属于一条消息：没有两条不同的消息用同一个 KEK 保护，一条消息的 KEK 也从不等于另一条消息的 KEK	无	成立	中	是
X02	**单条消息的临时泄露不会扩散。** 一条消息的临时值被披露后，其他每条消息的 KEK、CAEK 和内容都仍然保密	单条消息的一个临时值：该消息的成对 KEK（从线上被包裹的密钥即可推出 CAEK），或 CAEK 本身，或发起方在封装中使用的随机值。模型给出三者中最弱的一个，即一条消息的 KEK，因此在那里得到的答案对其余两者同样成立	成立	中	是
X03	**内容的数据源认证。** 除创建消息的一方之外，任何一方都不能构造出接收者会接受的消息：每条被接受的内容和认证属性都出自发起方的一次运行，因此接收者知道是哪一方发送的	无	攻击成立（文稿已写明）	中	是
X04	**接收方至多接受一条消息一次。** 接收者已接受过一次的 AuthEnvelopedData 不会被第二次当作新消息接受	无	不成立（1 个攻击）	低	否
X05	**每个 CAEK 仅使用一次 AEAD nonce。** 同一个内容认证加密密钥不会两次使用同一个 nonce 值	无	成立	低	是

“攻击成立（文稿已写明）”：证明器找到了这个攻击，而文稿自己已经写明它存在。这条性质不成立；这是已知结果，不算发现（附录 B）。

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, X03, X05.

文稿没有提到、由分析者按性质目录补充的性质（见附录 D）：X04 (R02).

所提的问题，以及分析者的回答：

问题	回答	依据
Q1: CEK/明文保密	在理想密码下，网络上的攻击者无法触及内容密钥和明文；但事后读取接收者 ML-KEM 私钥的人，或获知该消息密钥加密密钥的人，都能得到它们。	K02.8, C01, C04
Q2: recipient ID、KEM ct、KDF otherInfo、UKM 与算法标识绑定	被接受的密钥绑定到 KEM 密文、绑定到 wrap、kekLength 和 ukm，也绑定到 KEM 和 KDF 标识，因此替换其中任何一个都会使接收者拒绝该消息；接收者标识只是路由该条目的选择符，并未被绑定。	A08.9, A08
Q3: 篡改包裹 CEK/AEAD attributes 不能输出伪造的已验证内容	篡改会失败：修改或重组一条已产生的消息，会破坏密钥派生或标签。伪造内容则不会失败，因为任何持有接收者公钥的一方都能构造出接收者会验证并接受的消息，所以不能把验证理解为发送者认证。	D01.10, X03

4 本次分析的局限

这里的结果是对协议模型的结果：它说明规定的消息和检查给出了什么、未能给出什么，其中密码学视为理想，而不涉及任何实现。

按分析者的表述，本次分析不涵盖以下内容：

- ML-KEM 是一个符号化的项对，带有一条正确性规则。RFC 9935 第 9 节讨论的密钥与密文格式的绑定性质，以及任何具体的 ML-KEM 代数，都不在模型范围内。
- HKDF、AES-Wrap 和认证加密都是理想的。这里的任何结论都不涉及它们的计算强度、标签或密钥长度，也不涉及 nonce 误用行为。
- 随机性的质量没有被建模。密钥或封装值使用弱伪随机生成器的情况，即 RFC 9936 第 4 节和 RFC 9629 第 7 节所警告的情形，在本模型中无法被证明能破坏任何东西。
- 旁路信道、封装与解封装过程中的中间值，以及实现验证，都不在符号化模型范围内。
- 模型中只有一个接收者。多个接收者的 RecipientInfo 条目集合，以及选择恶意 KEM 密钥对的接收者，都没有被表示。
- 证书处理被视为已经完成：接收者的公钥被假定为其证书所携带的公钥，并且被假定已与其绑定。
- 释放顺序，即在校验标签之前不释放任何明文，在模型中是结构性的，因为获取内容的唯一途径是通过已验证的解密；它没有被作为一条单独的性质来证明。
- 不需要有界模型：没有任何目标依赖于计数器、被替换的密钥或其他跨步骤变化的值。

宿主记录的内容：

- 没带对照查询的证明（附录 C）：A08: rid_selector。对照查询是这条性质的一个必然不成立的变体；没有它，就没有东西表明这条性质照现在的写法在模型里有可能不成立。

每条性质还有哪些未决之处，以及怎样才能定论，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，不是证明器的答案：这些答案对协议意味着什么，解读依据什么，能否按原样使用，以及它有多大价值。

K01: 私钥、KEK、CAEK 的保密性

攻击者在不知道接收者 ML-KEM 私钥、KEK、CEK/CAEK 的情况下，不能推出这些密钥中的任何一个。

在攻击者只能访问网络的情况下，无论消息数量多少，私钥、成对密钥加密密钥和内容密钥都不会落入攻击者手中。一旦把相应的值交出去，对照查询就会失败，因此保密性依赖于从共享秘密经 HKDF 到 AES-Wrap 的链条，而不是依赖于模型中某个不可能发生的步骤。

- 结果：Holds；是否回答了所问的问题：是；可带保留地使用
- 价值中：它确认了一项核心保证，但只是针对理想原语，而这本来就是文稿所假定的。
- 文稿中已陈述：是。文稿中的位置：main:325, rfc9629:568
- 依据：ML-KEM 是一个带标签的项对，带有一条正确性规则；密钥或密文格式的绑定性质未被建模；HKDF、AES-Wrap 和认证加密是理想函数，因此这不是关于其计算强度的主张；KEK 与 CAEK 的保密性是对发送方的派生提出的；接受消息的接收者会派生出相同的值，字段绑定的答案表明了这一点；模型中只有一个接收者和一种消息布局

K02: 明文内容保密性

除发起方与目标接收者外，任何一方都不能从 AuthEnvelopedData 中推出被加密内容的明文。

在运行进行期间，无论发送多少条消息，网络上的攻击者都无法推出消息的明文。对照查询表明该查询是有意义的：一旦把接收者的私钥交出去，同一个攻击者就能恢复明文。

- 结果：Holds；是否回答了所问的问题：是；可带保留地使用
- 价值中：这是一项核心保密性保证，针对理想密码得到确认，且没有任何简化触及内容的加密。
- 文稿中已陈述：是。文稿中的位置：main:333, rfc5083:105
- 依据：认证加密是理想的，因此另一种长度的标签或密钥，或 nonce 误用，都不在模型范围内；内容是每个会话都携带的同一个固定私有值

D01: 内容与属性的一致性

接收者接受的每条已验证内容与认证属性，都对应发起方真实发送的同一内容与属性，未被攻击者替换或修改。

标签确实把认证属性绑定到了内容上，因此一条已经产生的消息无法被篡改，其属性也无法被重新组合后仍然通过验证。接收者无法做到的是分辨出是谁产生了这条消息：任何持有其公钥的一方可为它封装、构造自己的内容密钥，接收者会接受结果。

- 结果：攻击成立（文稿中已陈述）；是否回答了所问的问题：是；可带保留地使用
- 价值中：它恰好表明了共识在哪里停止：值受到保护，发送方身份并未确立，这正是文稿所陈述的。
- 文稿中已陈述：是。文稿中的位置：rfc9629:563, rfc5652:2864
- 依据：认证加密是理想的，其密钥被视为安全；接收者的公钥是公开的，正如文稿所意图；模型中只有一个发起方角色，因此就模型所能看到的而言，所有发送方都是同一方；因此失败的主张是目标在身份

层面的解读
- 未决：应用程序是否被允许把验证成功当作其对端写入了该消息；文稿禁止这种解读，但该字段存在，应用程序可能依赖它

A08: KEMRecipientInfo 字段的绑定

接收者接受的 KEK 与 CEK 绑定到发起方实际选择的接收者标识、KEM 密文、KDF 上下文（wrap、kekLength、ukm）与算法标识，任一字段被替换都会导致验证失败。

接收者派生出的密钥加密密钥被绑定到 KEM 密文、wrap 算法、密钥长度和用户密钥材料，以及 KEM 与 KDF 标识符，因为所有这些都参与了派生。替换其中任何一个都会使解封装失败。接收者标识只是一个选择器：接收者处理的是指名自己证书的那一条目，并不对该标识做密码学绑定。

- 结果：Holds；是否回答了所问的问题：是；可带保留地使用
- 价值中：它确认了文稿所要求的字段绑定，唯一的细微之处在于 rid 是选择器而非被绑定的值。
- 文稿中已陈述：是。文稿中的位置：rfc9629:349, rfc9629:287
- 依据：文稿所要求的 kekLength 与 wrap 算法之间的一致性检查没有被建模为一项单独检查；被替换的长度改由派生过程捕获；rid 被建模为接收者自己的标识，因此选择是结构性的，其对照查询只表明另一个标识不被接受；ASN.1 编码被理想化为元组

A05: 成对 KEK 与 CAEK 的隔离

对每个接收者，KEK 与 CEK 只对发起方和该接收者可得，其他实体不能推出。

成对密钥加密密钥和内容密钥只对发起方和被寻址的接收者可得。另一方，包括持有自己 ML-KEM 密钥对的一方，无法派生出它们；一条消息的所有接收者按设计共享内容密钥，因此隔离性对成对密钥以及其他消息的内容成立，泄漏隔离的答案覆盖了这一点。

- 结果：Holds；是否回答了所问的问题：是；可带保留地使用
- 价值中：它确认了文稿所依赖的成对构造；目标中多接收者的一面是由论证而非模型来回答的。
- 文稿中已陈述：是。文稿中的位置：rfc9629:97, rfc9629:574
- 依据：模型中只有一个接收者；在模型中另一个接收者处于网络攻击者的能力范围之内，因为它可以选择自己的密钥对；一条消息各接收者共享内容密钥这一点未被建模

C01: 针对私钥泄露的前向保密

在会话完成且临时密钥擦除之后泄露接收者 ML-KEM 私钥，攻击者仍不能推出此前已完成会话的 CEK 与内容。

没有前向保密。接收者的密钥是静态的，记录下来的 KEM 密文足以恢复共享秘密，因此事后读到私钥的攻击者可以解封装已经发送的每一条消息，派生出密钥加密密钥并解封装内容密钥。擦除发起方的临时值，即文稿所要求的做法，在接收者一侧保护不了任何东西。

- 结果：攻击成立（文稿中已陈述）；是否回答了所问的问题：是；可使用
- 价值中：文稿已陈述私钥泄露会造成这一损失；分析现在给出了确切的步骤，并确认擦除并不能阻止它。
- 文稿中已陈述：是。文稿中的位置：main:329, rfc9629:569
- 依据：泄漏被建模为在运行之后读取密钥；擦除本身除该顺序之外未被建模；接收者不保留任何可保护已完成消息的逐消息状态

C04: 临时组件泄露的后果

在指定临时组件（KEK、CEK/CAEK、封装随机值）之一泄露时，检查其余保密与完整性目标是否仍成立。

泄露一条消息的密钥加密密钥会给出该消息的内容密钥及其内容，因为被包装的密钥在网络上传输。泄露内容密钥或封装随机值至少同样有力，并给出相同的结果。损害局限于该消息：相邻消息保留各自的密钥。

- 结果：攻击成立（文稿中已说明）；是否回答了所问：是；可用但有注意事项
- 价值低：它来自攻陷所交出的东西，而不是来自协议的性质。
- 文稿中已说明：是。文稿中的位置：main:333, rfc9629:571
- 依据：只对密钥加密密钥的泄露建模；目标的另外两个组成部分由同一答案覆盖，因为它们各自都能推出该消息的内容密钥；泄露下的完整性在这里不增加任何内容，因为模型已经允许任何持有接收者公钥的参与方发送它能接受的消息

K02.8: CEK 与明文保密性

攻击者不能推出 CEK 或被加密内容的明文。

内容密钥和明文始终不在网络攻击者的触及范围之内，一旦接收者的私钥或消息的密钥加密密钥被泄露，它们就随之失去。

- 结果：成立；是否回答了所问：是；可用但有注意事项
- 价值中：它用一个证明回答了用户的第一个问题，并给出了该性质不再成立的条件。
- 文稿中已说明：是。文稿中的位置：main:333, rfc5083:105
- 依据：理想认证加密；内容是单个固定的私有值

A08.9: KEMRecipientInfo 字段绑定

接收者接受的 KEK 与 CEK 绑定到发起方实际选择的接收者标识、KEM 密文、KDF 上下文与算法标识。

接收者接受的密钥绑定到 KEM 密文、派生上下文和算法标识，因此字段无法在其下被调换。接收者标识只负责选择条目，并不参与绑定；密钥绑定来自 KEM 密文和接收者的私钥。

- 结果：成立；是否回答了所问：是；可用但有注意事项
- 价值中：它回答了用户的第二个问题，包括那个只作选择器、不作绑定的字段。
- 文稿中已说明：是。文稿中的位置：rfc9629:349
- 依据：密钥长度一致性检查被派生绑定取代；上下文的 DER 编码被理想化；选择器部分的答案基于结构，其对照查询只表明另一个标识会被拒绝

D01.10: 内容与属性一致性

接收者接受的已验证内容与认证属性对应发起方真实发送的同一内容与属性，未被篡改。

对已产生消息的改动和重组会被标签和密钥派生发现，因此接收者接受的值就是发送出去的值。归属则完全不会被发现：接收者会接受任何持有其公钥的参与方所写的内容。

- 结果：攻击成立（文稿中已说明）；是否回答了所问：是；可用但有注意事项
- 价值中：它从两个方向回答了用户的第三个问题：篡改被排除，消息伪造没有被排除。
- 文稿中已说明：是。文稿中的位置：rfc9629:563, rfc5083:119

- 依据：理想认证加密；模型中只有一个发起方角色，因此不成立的那种解读是关于发送者是谁的解读

X01: 每条消息使用新的 KEM 共享密钥

The KEM shared secret and the KEK derived from it belong to one message only: no two distinct messages are protected with the same KEK, and the KEK of one message never equals the KEK of another

每条消息都使用自己的 KEM 共享密钥，因此它的密钥加密密钥也是自己的，消息之间不共享任何密钥。当密钥与其消息之间的关联被去掉时，对照查询失败，这表明该查询需要每条消息的新鲜共享密钥。

- 结果：成立；是否回答了所问：是；可用但有注意事项
- 价值中：它确认了文稿用大写字母写出的 requirements，其对照查询表明正是这一要求支撑了该结果。
- 文稿中已说明：是。文稿中的位置：rfc9629:574
- 依据：假定再次执行封装时 KEM 会产生新的共享密钥，这正是该答案所检查的要求

X02: 单条消息的临时泄露不会扩散

Disclosure of an ephemeral value of one message leaves the KEK, the CAEK and the content of every other message secret

一条消息的密钥加密密钥泄露不会扩散：另一条消息的密钥和内容密钥仍然保密，因为每条消息都用各自的共享密钥派生自己的密钥。对照查询是对被泄露密钥的那条消息提出同一查询，在那里它不成立。

- 结果：成立；是否回答了所问：是；可用但有注意事项
- 价值中：它确认了文稿对临时泄露所承诺的隔离，并把机制明确写出。
- 文稿中已说明：是。文稿中的位置：main:333, rfc9629:571
- 依据：只对一条被泄露的消息建模；用两条独立消息代表任意数量

X03: 内容的数据源认证

A party other than the one that created a message cannot produce a message the recipient accepts: every accepted content and authenticated attribute was sent by a run of the originator, so the recipient knows which party sent it

接收者没有证据表明消息是谁写的。任何读到其证书的参与方都能为它封装，并产生一个能通过验证的完整 AuthEnvelopedData，因此验证成功只说明有人运行了协议，从不说明是哪个参与方。文稿明确说明了这一点，因此它们并不把解密成功称为发送者认证。

- 结果：攻击成立（文稿中已说明）；是否回答了所问：是；可用
- 价值中：它用确切的攻击复现了文稿记载的限制，这正是实现者在依赖验证之前需要的。
- 文稿中已说明：是。文稿中的位置：rfc9629:563, rfc5652:2864
- 依据：接收者的公钥是公开的，且证书校验被视为已完成；模型中不存在签名或其他发起方凭证，这与任务设定一致

X04: 接收方至多接受一条消息一次

An AuthEnvelopedData that the recipient has accepted once is not accepted a second time as a new message

没有任何机制阻止同一条消息被接受两次。接收者不记录已接受过什么，内容类型也不携带它所检查的计数器、时间戳或挑战值，因此一个被记录下来的 AuthEnvelopedData 再次送达时会被再次解密并接受，内容和属性都相同。只有应用才能区分首次送达和重复送达。

- 结果：不成立（1 个攻击）；是否回答了所问：是；可用但有注意事项
- 价值低：这是无状态内容类型的后果，而不是其设计的缺陷；文稿完全没有讨论它，因此它是这里唯一尚未记载的结果。
- 文稿中已说明：否
- 依据：模型中的接收者没有状态，这正是内容类型所定义的；该轨迹通过一条攻击者构造的消息到达第二次接受，但接收者无法将其与截获的真实消息区分开
- 待定：是否应由应用提供新鲜性，例如检查一个签名时间属性，留给了使用该内容类型的配置文件
- 下一步：阅读使用该内容类型的应用配置文件；如果新鲜性由它负责，该结果就是预期内的，属于该配置文件。

X05: 每个 CAEK 仅使用一次 AEAD nonce

No nonce value is used twice with the same content-authenticated-encryption key

每条消息都在新生成的内容密钥下加密一次，因此同一个密钥下不会出现 nonce 重复。当与某个密钥的关联被去掉时，对照查询失败，这表明该查询关注的就是这一关联。

- 结果：成立；是否回答了所问：是；可用但有注意事项
- 价值低：它由构造方式直接成立，而且文稿已经要求每个内容使用新的内容密钥。
- 文稿中已说明：是。文稿中的位置：rfc5083:325
- 依据：每个内容密钥只加密一次是构造方式的性质，而不是某项检查；若某个配置文件在一个密钥下加密多个内容，就需要另行规定 nonce 规则

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、是否检查了对照查询，以及各个攻击及分析者对每个攻击的判定。

目标	级别	界	查询	对照	攻击	主张
K01	proved		sk_secret	checked		3 proved
K02	proved		content_secret	checked		1 proved
D01	proved		mod_binding	checked	agreement (holds_in_source)	2 proved, 1 attack
A08	proved		mod_binding	4 of 5		5 proved
A05	proved		kek_secret	checked		2 proved
C01	reachable		sk_ctl		c01_caek (holds_in_source); content_ctl (holds_in_source)	2 attack
C04	reachable		reach_kek		c04_caek (holds_in_source); c04_content (holds_in_source)	2 attack
K02.8	proved		caek_secret	checked		2 proved
A08.9	proved		mod_binding	checked		2 proved
D01.10	proved		attr_binding	checked	agreement (holds_in_source)	1 proved, 1 attack
X01	proved		kek_fresh	checked		1 proved
X02	proved		x02_iso	checked		1 proved
X03	reachable		run		agreement (holds_in_source)	1 attack
X04	reachable		run		replay (holds_in_source)	1 attack
X05	proved		nonce_once	checked		1 proved

逐条主张，每条主张按分析者的措辞给出；每个答案都是证明器给出的：

目标	主张	攻击者	回答	查询	对照
K01	接收方的静态 ML-KEM 私钥无法被攻击者推导出来，即使攻击者拥有网络且不持有任何一方的密钥。	network	proved	sk_secret	checked
K01	发送方运行所推导出的成对 KEK 无法被网络攻击者推导出来，前提是尚未披露任何临时值。	network	proved	kek_secret	checked
K01	发送方运行所生成的 CAEK 无法被网络攻击者推导出来，前提是尚未披露任何临时值。	network	proved	caek_secret	checked
K02	在运行进行期间，消息的明文内容无法被网络攻击者推导出来。	network	proved	content_secret	checked
D01	改动某一方所生成消息的 KEM 密文 kemct、kem 或 kdf 标识符，或 KDF 上下文（wrap、kekLength、ukm），会使接收方推导出不同的 KEK，因此解封装失败，消息被拒绝。	network	proved	mod_binding	checked
D01	改动某一方所生成消息的 authenticated attributes，或将一条消息的属性与另一条消息的密文重新组合，都会被拒绝：接收方接受的属性正是该方所发送的。	network	proved	attr_binding	checked

目标	主张	攻击者	回答	查询	对照
D01	接收方所接受的每一项内容与 authenticated attribute 都由发起方的一次运行产生，因此接收方知道是谁发送的。攻击者只需要接收方的公钥。	`network`	attack (holds_in_source)	agreement	
A08	替换所生成消息的 KEM 密文 kemct，或替换 kem 与 kdf 算法标识符，会被拒绝：接收方推导出不同的共享秘密与 KEK，因此解封装失败。	`network`	proved	mod_binding	checked
A08	替换所生成消息 KDF 上下文中的 wrap、kekLength 或 ukm，会被拒绝：KDF 上下文参与 KEK 推导，因此接收方的 KEK 不同，解封装失败。	`network`	proved	mod_binding	checked
A08	authenticated attributes 位于 AEAD 输入之内，因此替换或重新组合它们会使标签校验失败，消息被拒绝。	`network`	proved	attr_binding	checked
A08	rid 是选择器，不是密码学绑定：接收方只处理 rid 指向自身证书的 RecipientInfo，因此被替换的 rid 会使其跳过该条目，而不是接受它。	`network`	proved	rid_selector	none
A08	rid 是选择器，不是密码学绑定：接收方只处理 rid 指向自身证书的 RecipientInfo，因此被替换的 rid 会使其跳过该条目，而不是接受它。对照查询表明，当针对另一接收方的标识符询问同一查询时，该查询不成立。	`network`	proved	rid_selector	checked
A05	成对 KEK 仅对发起方和被寻址的接收方可用：网络上的攻击者即使自身持有另一方的 ML-KEM 密钥对，也无法推导出它。	`network`	proved	kek_secret	checked
A05	为该接收方封装的 CAEK 仅对发起方和该接收方可用；网络上的其他任何一方，包括持有不同 ML-KEM 密钥对的一方，都无法推导出它。	`network`	proved	caek_secret	checked
C01	在泄漏发生之前已完成的消息，其 CAEK 在接收方的 ML-KEM 私钥被披露且临时密钥被擦除之后仍保持机密。	`mk_sk_leak`	attack (holds_in_source)	c01_caek	
C01	在泄漏发生之前已完成的消息，其内容在接收方的 ML-KEM 私钥被披露之后仍保持机密。	`mk_sk_leak`	attack (holds_in_source)	content_ctl (the control of content_secret)	
C04	在一条消息的 KEK 被披露的情况下，该消息的 CAEK 仍保持机密。攻击只需要来自线路上的 KEK 和封装密钥；披露 CAEK 或封装随机性至少具有同样的强度。	`ephemeral_leak`	attack (holds_in_source)	c04_caek (the control of x02_iso)	
C04	在一条消息的 KEK 被披露的情况下，其内容仍保持机密。	`ephemeral_leak`	attack (holds_in_source)	c04_content	
K02.8	CAEK（即认证加密的 CEK）无法被网络上的攻击者推导出来。	`network`	proved	caek_secret	checked
K02.8	加密内容的明文无法被网络上的攻击者推导出来。	`network`	proved	content_secret	checked
A08.9	接收方获得的 KEK 与 CAEK 绑定到 kemct、KDF 上下文（wrap、kekLength、ukm）以及 kem 与 kdf 标识符：任何替换都会使接收方拒绝该消息。	`network`	proved	mod_binding	checked

目标	主张	攻击者	回答	查询	对照
A08.9	rid 标识接收方条目，且不受密码学绑定：接收方按自身标识符选择条目，密钥的绑定改为来自 kemct 及其私钥。	`network`	proved	rid_selector	checked
D01.10	在接受所生成消息的内容与 authenticated attributes 时，无法在保持接收方验证成功的同时对其加以改动或重新组合。	`network`	proved	attr_binding	checked
D01.10	所接受的内容与属性来自发起方的真实运行，因此接收方可以归属它们。攻击者只需要接收方的公钥，就能让接收方接受其自行选择的内容。	`network`	attack (holds_in_source)	agreement	
X01	任意两条不同的消息都不会用同一个 KEK 保护：每条消息都使用新的 KEM 共享秘密，因此一条消息的 KEK 与 CAEK 绝不会用于另一条消息。	`network`	proved	kek_fresh	checked
X02	披露一条消息的 KEK 不会影响其他每条消息的 KEK 与 CAEK 的机密性：第二条消息的泄漏不会波及 KEK 未被披露的那条消息。	`ephemeral_leak`	proved	x02_iso	checked
X03	接收方确信所接受的消息由它认为正在通信的发起方创建，且其他任何一方都无法生成它接受的消息。	`network`	attack (holds_in_source)	agreement	
X04	接收方已经接受过一次的 AuthEnvelopedData 不会被再次作为新消息接受。	`network`	attack (holds_in_source)	replay	
X05	在同一个 CAEK 下，任何 nonce 值都不会被使用两次：每条消息都只在其新生成的 CAEK 下加密一次。	`network`	proved	nonce_once	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，并将每个性质交给 ProVerif 验证；第 3 节中的每个结果都是证明器的回答。

ProVerif+ Agent 会话 session-20261114T222716，2026-11-14。状态：已完成。模型：deepseek-v4-1-flash-261017。步数：37，证明器 8 s，总耗时 14 min，成本 0.121 USD。

ProVerif：共运行 96 次证明，96 次得出判定，0 次无答案，耗时 4 s，记录 45 个回答。

分析了什么

协议	CMS ML-KEM AuthEnvelopedData
版本	rfc9936
分析范围	固定 ML-KEM-768 KEMRecipientInfo、HKDF、单接收者 AuthEnvelopedData，AEAD 理想化并显式关联 authenticated attributes；接收者公钥已验证且不包含发送者签名。该过程是一次加密消息，不把解密成功当发送者认证。
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	130 个编号原文单元（28306 字节），来自下文所列各节。本报告中诸如 s0001 的引用即指这些原文单元之一

分析所依据的文稿各节如下：

文稿	章节	标题
main	2	ML-KEM 算法在 CMS 中的使用
main	2.1	RecipientInfo 约定
main	2.2	底层组件
main	2.2.1	基于 HKDF 的密钥派生函数的使用
main	2.3	证书约定
main	2.4	SMIME Capabilities 属性约定
main	4	安全考虑
rfc9629	2	KEM 处理概述
rfc9629	3	KEM 接收者信息
rfc9629	5	密钥派生
rfc5083	2	Authenticated-Enveloped-Data 内容类型
rfc5083	2.1	AuthEnvelopedData 类型
rfc5083	2.2	认证与加密过程
rfc5083	2.3	密钥加密过程

协议的目的，按原文步骤在文稿中的定位：固定 ML-KEM-768 KEMRecipientInfo、HKDF、单接收者 AuthEnvelopedData，AEAD 理想化并显式关联 authenticated attributes；接收者公钥已验证且不包含发送者签名。该过程是一次加密消息，不把解密成功当发送者认证。

对照文稿定性质

在编写任何查询之前，分析者将草拟的性质清单与文稿自身所述协议目的进行对照。名称、性质所涉及的对象以及文稿的说法均属分析者的解读；所引位置在文稿中确实存在。

ID	简称	对象	文稿怎么说	出处
K01	私钥、KEK、CAEK 的保密性	接收者的 ML-KEM 私钥、成对 KEK 以及内容认证加密密钥（及其所代表的 AEAD 密钥）对网络上的攻击者保密	RFC 9936 第 4 节要求实现方保护 ML-KEM 私钥、KEK、CEK、MAK 和 CAEK，并说明了这些信息泄露会危及什么；RFC 9629 第 2 节说明 KEK 是成对的，第 7 节重申了保护要求	main:325, rfc9629:568, rfc9629:97, s0034, s0043, s0044, s0051, s0056
K02	明文内容保密性	发起方认证并加密的内容只能由发起方和指定接收者读取	RFC 9936 第 4 节将 CAEK 的泄露列为会危及相关联的加密内容；RFC 5083 第 2 节说明内容经过认证和加密，接收者必须在释放明文之前验证完整性	main:333, rfc5083:85, s0034, s0091, s0105, s0106
D01	内容与属性的一致性	接收者验证标签后所接受的内容：所接受的内容和 authenticated attributes 确实是某一方真实发送且未被修改的	RFC 5083 第 2 节说明标签直接覆盖 authenticated attributes 和内容，接收者必须在释放明文之前验证；RFC 9629 第 7 节说明 KEM 不提供数据源认证，因此内容以来自未知来源的完整性交付	rfc5083:119, rfc9629:563, s0102, s0105, s0106, s0119
A08	KEMRecipientInfo 字段的绑定	接收者获得的 KEK 和 CAEK 绑定到 rid、KEM 密文 kemct、KDF 上下文（wrap、kekLength、ukm）以及发起方所选的 kem 和 kdf 算法标识符	RFC 9629 第 5 节说明 info 输入由 KEMRecipientInfo 的字段 wrap、kekLength 和 ukm 构建，并且所有 KDF 输入都必须影响输出；RFC 9936 第 2.1 节固定了算法标识符，RFC 9629 第 3 节要求 kekLength 与密钥加密算法保持一致	rfc9629:349, rfc9629:287, s0009, s0011, s0013, s0014, s0081, s0082, s0083, s0084, s0085, s0086, s0087, s0088, s0089
A05	成对 KEK 与 CAEK 的隔离	接收者的成对 KEK 及其所保护的 CAEK 仅对发起方和该接收者可用	RFC 9629 第 2 节说明 KEM 和 KDF 建立成对的对称 KEK，用于加密发起方生成的 CEK 或 CAEK，第 7 节要求为每个接收者使用新的 KEM 共享密钥	rfc9629:97, rfc9629:574, s0044, s0049, s0050, s0051, s0054, s0055, s0056
C01	针对私钥泄露的前向保密	已完成的某条消息的 CAEK 和内容，面对在消息发送完毕且临时密钥已被清除之后获得接收者 ML-KEM 私钥的攻击者	RFC 9936 第 4 节说明除私钥外均为临时密钥，使用后必须清除；ML-KEM 私钥的泄露可能导致用该密钥保护的所有消息被危及；目标追问清除是否保护了过往消息	main:329, rfc9629:569, s0034
C04	临时组件泄露的后果	单条消息的某个临时组件泄露时，同一协议的其他保密性和完整性目标	RFC 9936 第 4 节说明 KEK、CEK 或 CAEK 的泄露可能危及相关联的加密内容并允许修改相关联的认证内容，封装随机值必须选择得当；RFC 9629 第 7 节将此类危及限制在单条消息	main:333, main:349, rfc9629:571, s0034, s0036
K02.8	CEK 与明文保密性	消息的 CAEK 和明文，面对网络上的攻击者	与 K02 相同的目标，由用户问题 Q1 提出：RFC 9936 第 4 节和 RFC 5083 第 2 节	main:333, rfc5083:105, s0034, s0091, s0105, s0106
A08.9	KEMRecipientInfo 字段绑定	所接受的 KEK 和 CAEK 及其与 rid、kemct、KDF 上下文和算法标识符的绑定，由用户问题 Q2 提出	与 A08 相同的目标：RFC 9629 第 5 节和第 3 节	rfc9629:349, rfc9629:287, s0081, s0082, s0083, s0087, s0088, s0089

ID	简称	对象	文稿怎么说	出处
D01.10	内容与属性一致性	所接受的内容和 authenticated attributes，由用户问题 Q3 提出	与 D01 相同的目标：RFC 5083 第 2 节和 RFC 9629 第 7 节	rfc5083:119, rfc9629:563, s0102, s0105, s0106, s0119
X01	每条消息使用新的 KEM 共享密钥	发起方与接收者之间的成对共享密钥和 KEK	RFC 9629 第 7 节要求为 KEMRecipientInfo 结构的每个接收者使用新的 KEM 共享密钥，禁止将 IKM 复用于任何其他目的，并要求在使用后丢弃共享密钥和 KEM 随机数	rfc9629:574
X02	单条消息的临时泄露不会扩散	除被危及消息之外的其他消息的 KEK、CAEK 和内容	RFC 9936 第 4 节将 KEK、CEK 或 CAEK 的泄露仅与相关联的加密内容联系起来；RFC 9629 第 7 节说明 KEK、CEK 或 CAEK 的危及可能导致单条消息的加密内容泄露	main:333, rfc9629:571
X03	内容的数据源认证	接收者所看到的发起方身份，以及其所接受的内容和 authenticated attributes	RFC 9629 第 7 节声明 KEM 算法不提供数据源认证，因此内容以来自未知来源的完整性交付；RFC 5652 第 14 节对不认证发起方的密钥管理技术作了相同声明	rfc9629:563, rfc5652:2864
X04	接收方至多接受一条消息一次	接收者对一条消息的接受，面对记录该消息并再次投递的攻击者	关于消息级重放保护的目录条目 R02；本分析的文稿未对 AuthEncryptedData 声明任何新鲜性或防重放机制	
X05	每个 CAEK 仅使用一次 AEAD nonce	认证加密算法在同一个 CAEK 下使用的 nonce	RFC 5083 第 3 节警告在同一密钥下重用计数器块会使保密性失效，并说明每条内容使用新的内容加密密钥、以及在算法标识符参数中携带不可预测的 nonce 可以避免该误用	rfc5083:325

分析者依据文稿补充的性质：X01、X02、X03、X05。

文稿里的安全表述，逐段去向

宿主将文稿中说明协议保护什么、防范什么的章节逐段列出，共 46 段；分析者对每一段说明了去向：定为性质、定为攻击者条件，或者说明为何两者都不选。选了 15 段，未选 31 段。选择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
P01 至 P02	main:319-323	没有陈述目标	两处都指向 RFC 9935 和 RFC 9629 的安全考虑，以及一个外部讨论列表；这些段落已在此处单独列出
P03	main:325-336	K01, C01, C04, `mk_sk_leak`, `ephemeral_leak`	
P04	main:338-339	没有陈述目标	指向 NIST SP 800-57 中关于密钥管理的内容
P05	main:341-356	C04	
P06	main:358-360	证明器问不了	一条关于实现如何使用封装和解封装中间值的规则；模型中没有中间值，而共享密钥和 KEK 的保密性由 K01 询问
P07	main:362-369	证明器问不了	时序和其他侧信道不在符号模型的范围内

段落	出处	定为	不选的原因
P08	main:371-374	证明器问不了	一条关于在多个方案中复用同一对 ML-KEM 密钥对的规则；这是跨协议的性质，而本模型只包含一个方案
P09	main:376-378	没有陈述目标	涉及形式化的实现验证（CMVP），这是一个保证流程，而不是消息流的性质
P10	rfc5083:308-310	没有陈述目标	指向 CMS（RFC 5652）的安全考虑；这些段落已在此处单独列出
P11	rfc5083:312-317	没有陈述目标	关于计数器模式保密性的背景介绍，与本协议的目标无关
P12	rfc5083:325-328	X05	
P13	rfc5083:330-337	X05	
P14	rfc5083:339-348	K01	
P15	rfc5083:350-355	不在范围内	警告 CMS 的可选消息摘要属性；认证属性的模型不包含该属性，同一段也已禁止包含它
P16	rfc5083:357-369	不在范围内	对未经请求的加密邮件进行垃圾邮件和钓鱼过滤属于部署实践，不是交换的性质
P17	rfc5652:2837-2839	没有陈述目标	描述 CMS 一般提供什么，与本内容类型的目标无关
P18	rfc5652:2841-2842	没有陈述目标	签名者私钥泄露；本协议不使用签名，且任务已固定不存在发送方签名，因此该段在此不陈述目标
P19	rfc5652:2844-2849	K01, C01, C04, `mk_sk_leak`, `ephemeral_leak`	
P20	rfc5652:2851-2855	D01, C04, `ephemeral_leak`	
P21	rfc5652:2864-2872	X03	
P22	rfc5652:2874-2877	没有陈述目标	关于认证数据内容类型中多方共享一个消息认证密钥；本协议没有共享 MAC 密钥，因此该段在此不陈述目标
P23	rfc5652:2879-2889	K01	
P24	rfc5652:2891-2903	证明器问不了	有效强度取两个算法中较弱者，这是关于原语强度的陈述，符号模型不度量这一点
P25	rfc5652:2905-2911	没有陈述目标	建议算法会随时间变弱，实现应保持模块化
P26	rfc5652:2917-2927	不在范围内	签名数据内容类型的计数器签名，本协议不使用
P27 至 P28	rfc5869:355-371	证明器问不了	HKDF 的密码学分析及其对哈希函数的依赖；模型将 HKDF 视为理想的密钥派生函数
P29	rfc8619:195-199	证明器问不了	HKDF 配合 SHA-256 的安全考虑涉及哈希函数的强度，而不是消息流
P30	rfc9629:542-543	没有陈述目标	指向 CMS（RFC 5652）的安全考虑，已单独列出
P31 至 P33	rfc9629:545-561	证明器问不了	KEM 的 IND-CCA2 安全性以及 KDF 与密钥加密算法安全级别的匹配，都是关于原语强度的陈述
P34	rfc9629:563-566	X03	
P35	rfc9629:568-572	K01, C01, C04, `mk_sk_leak`, `ephemeral_leak`	

段落	出处	定为	不选的原因
P36	rfc9629:574-587	X01, X02	
P37	rfc9629:589-598	K01	
P38 至 P40	rfc9629:600-620	证明器问不了	有效强度取两个算法中较弱者，这是关于原语强度的陈述
P41	rfc9629:622-629	没有陈述目标	建议算法会随时间变弱，实现应保持模块化
P42	rfc9935:406-407	没有陈述目标	指向 RFC 5280 的安全考虑
P43	rfc9935:409-411	K01, C01, X03, `mk_sk_leak`	
P44	rfc9935:413-421	证明器问不了	ML-KEM 的 MAL-BIND-K-CT 和 MAL-BIND-K-PK 绑定性质是 KEM 密钥和密文格式的代数性质；符号化处理将封装视为带标签的项对，无法表达这些性质；已在局限中注明
P45	rfc9935:423-431	K01	
P46	rfc9935:433-435	没有陈述目标	指向一份外部讨论文稿

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 10 条，没检查 25 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	X01	
A01	对端曾参与性（Aliveness）	X03	
A02	弱一致性认证	X03	
A03	非单射一致性认证	X03	
A04	单射一致性认证	X04	
A06	显式密钥确认	不适用	本协议是从发起方到接收方的一条消息，没有确认消息，因此其中没有任何内容涉及显式密钥确认
A07	身份—凭据—密钥绑定	不在范围内	证书与身份的绑定属于 X.509 验证；任务已固定接收方公钥为已验证，因此排除了这一部分
R01	认证／挑战新鲜性	不适用	本协议没有挑战、计数器或时间戳，其新鲜性无从检查
R02	消息级抗重放	X04	
R03	业务操作至多一次	不在范围内	业务操作的至多一次属于使用该内容类型的应用
R04	消息接受顺序／窗口安全	不适用	本协议没有序列号，也没有接受窗口
R05	抗降级／安全策略保持	A08	
C02	抗密钥泄露冒充（KCI）	不适用	本协议不认证任何对端，因此不存在可排除密钥泄露后冒充的对端
C03	已知密钥安全／泄露隔离	X02	
C05	泄露后安全恢复（PCS）	不适用	不存在通信可以恢复的会话或棘轮状态；每条消息相互独立，私钥是静态的
C06	注册凭据泄露后的增强认证	不适用	本协议没有口令，没有增强的口令认证交换，也没有注册记录

条目	目录里的性质	作为哪条性质检查	不检查的原因
C07	抗离线字典攻击	不适用	本协议没有口令，没有增强的口令认证交换，也没有注册记录
C08	指定原语失效下的目标保持	不适用	规范只固定了 ML-KEM，没有混合或回退组合，而 C08 所说的失效目标正是指这类组合
Z01	授权执行正确性	不在范围内	本协议是一个加密内容类型，不涉及授权、委托、撤销或租户隔离
Z02	委托权限不扩张	不在范围内	本协议是一个加密内容类型，不涉及授权、委托、撤销或租户隔离
Z03	授权与身份／会话／操作绑定	不在范围内	本协议是一个加密内容类型，不涉及授权、委托、撤销或租户隔离
Z04	撤销与过期生效	不在范围内	本协议是一个加密内容类型，不涉及授权、委托、撤销或租户隔离
Z05	跨租户／跨服务安全隔离	不在范围内	本协议是一个加密内容类型，不涉及授权、委托、撤销或租户隔离
S01	加密 nonce 唯一性	X05	
S02	重传／密钥重安装安全	不适用	本协议不安装密钥状态，不维护计数器，也没有可能被重传回滚的阶段或终止状态
S03	抗回滚与状态单调性	不适用	本协议不安装密钥状态，不维护计数器，也没有可能被重传回滚的阶段或终止状态
S04	跨阶段安全上下文一致性	不适用	本协议不安装密钥状态，不维护计数器，也没有可能被重传回滚的阶段或终止状态
S05	终止与旧状态失效	不适用	本协议不安装密钥状态，不维护计数器，也没有可能被重传回滚的阶段或终止状态
P01	身份匿名性	不适用	接收方通过证书引用识别，且本协议不认证任何人，因此既没有声称也没有要求任何匿名或隐藏身份
P02	会话不可关联性	不适用	接收方通过证书引用识别，且本协议不认证任何人，因此既没有声称也没有要求任何匿名或隐藏身份
P03	用户／凭据存在性隐藏	不适用	接收方通过证书引用识别，且本协议不认证任何人，因此既没有声称也没有要求任何匿名或隐藏身份
V01	条件性进展／错误恢复	不在范围内	进展、死锁和资源放大问题不在本任务的符号流分析范围内
V02	无非预期死锁	不在范围内	进展、死锁和资源放大问题不在本任务的符号流分析范围内
V03	受限资源滥用／抗放大	不在范围内	进展、死锁和资源放大问题不在本任务的符号流分析范围内

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 27 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 9 项，没设的 18 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	发起方	参与方	攻击者得不到更多	处于发起方位置的攻击者自己写内容；接收者无论如何都无法对发起方获得任何保证，而这正是数据来源认证目标所要求的
S02	接收者	参与方	`mk_sk_leak`	
S03	攻击者	参与方	攻击者得不到更多	网络上的攻击者就是所建模的攻击者；此条目命名的是攻击者本身
S04	目标接收者	参与方	`mk_sk_leak`	

ID	参与方或值	类别	攻击者条件	不设的原因
S05	其他接收者	参与方	按定义全部失守	一条消息的所有接收者按设计共享同一个 CAEK，因而共享内容（RFC 9629 第 2 节）；每个接收者的 KEK 保持成对，A05 对网络攻击者要求的正是这一点，而该攻击者本身可能就是另一个接收者
S06	接收者 ML-KEM 私钥	值	`mk_sk_leak`	
S07	KEK	值	`ephemeral_leak`	
S08	CEK/CAEK	值	`ephemeral_leak`	
S09	消息认证密钥	值	`ephemeral_leak`	
S10	被加密内容	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S11	认证属性	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S12	包裹的 CEK	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S13	接收者标识 rid	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S14	KEM 密文 kemct	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S15	KDF 上下文	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S16	算法标识 kem/kdf	值	攻击者得不到更多	内容、认证属性（作为 AAD 明文发送）、包裹的密钥、rid、kemct、KDF 上下文和算法标识都在线路上，因此持有它们的攻击者并不拥有网络攻击者所缺少的任何东西
S17	历史会话的 CEK/CAEK	值	`mk_sk_leak`	
S18	历史会话内容	值	`mk_sk_leak`	
S19	封装随机值	值	`ephemeral_leak`	
S20	内容	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中
S21	内容明文	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中
S22	rid	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中
S23	kemct	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中
S24	wrap	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中
S25	kekLength	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中

ID	参与方或值	类别	攻击者条件	不设的原因
S26	ukm	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中
S27	kem/kdf 标识	值	攻击者得不到更多	攻击者已经持有的消息的内容、它创建的消息的明文、rid、kemct、wrap、kekLength、ukm 和算法标识都是公开的或已在它手中

每条性质在各攻击者条件下的答案

攻击者由分析者设定，在编写查询之前就定了；每个答案都记录在其中一个攻击者之下。单元格是证明器在该攻击者下对该性质给出的答案；“无答案”是该性质下证明器让某一主张未决或什么也没记录时的攻击者集合；空单元格是分析者未为该性质设定的攻击者。

攻击者	持有什么	依据	出处
`network`	仅网络：它读取、丢弃、更改和发送消息，不持有任何一方的秘密		
`mk_sk_leak`	接收者的静态 ML-KEM 私钥，在消息发送之后、发起方的临时密钥被擦除之后读取	RFC 9936 第 4 节：ML-KEM 私钥的泄露可能导致使用该密钥保护的所有消息被入侵；RFC 9629 第 7 节：KEM 私钥的泄露可能导致使用该 KEM 私钥保护的所有内容被泄露	main:329, rfc9629:568, s0034
`ephemeral_leak`	单条消息的一个临时值：该消息的成对 KEK（由它结合线路上的包裹密钥可推出 CAEK），或 CAEK 本身，或发起方在封装中使用的随机值。模型给出三者中最弱的一个，即一条消息的 KEK，因此在此找到的答案对其他两者同样成立	RFC 9936 第 4 节：KEK、CEK 或 CAEK 的泄露可能导致相关加密内容被入侵，KEK 或 CAEK 的泄露可能允许修改相关认证内容；RFC 9629 第 7 节：KEK、CEK 或 CAEK 的泄露可能导致单条消息的加密内容被泄露	main:333, rfc9629:571, s0034, s0036

ID	`network`	`mk_sk_leak`	`ephemeral_leak`
K01	成立		
K02	成立		
D01	不成立		
A08	成立		
A05	成立		
C01		不成立	
C04			不成立
K02.8	成立		
A08.9	成立		
D01.10	不成立		
X01	成立		
X02			成立
X03	不成立		
X04	不成立		
X05	成立		

原文和目标

原始出处：130 个原文单元，28306 字节，冻结于 source-delivery-01.json。 目标：7 个已批准（K01、K02、D01、A08、A05、C01、C04）；提出的问题：3 个。 目标列表事先按草稿获批准，使用 --approve-targets。

宿主按其文档化的默认值做出的决定（任务 `decisions: auto`），供你核对：

- approve_package: 批准源文包 protocol-b9eca6830064aa836aecf42150e9

本步骤的自述：1 次请求（1 次失败），0.000 USD，24868 个 token 输入、3851 个输出，22 秒。

关于证明器和模型的说明

分析者为目标指定了：

步骤	目标	特征	原因
28	K01, K02, D01, A08, A05, C01, C04, K02.8, A08.9, D01.10, X01, X02, X03, X04, X05	plain_primitives	所有答案仅基于简单原语：ML-KEM 封装和解封装是带一条正确性规则的标记项对，HKDF 是共享秘密和 DER 上下文的理想密钥派生函数，AES-Wrap 和 AEAD 是理想的认证构造，标签覆盖这些属性。没有任何目标依赖计数器或被替换的密钥、异或或 Diffie-Hellman 代数，因此 model.pv 不需要有界展开。

分析者的结束说明

该模型是 CMS ML-KEM AuthEnvelopedData 正常流程的 ProVerif 模型：一个发起方、一个接收者，已发布的经认证的 ML-KEM-768 公钥，ss → KEK = HKDF(ss, DER(wrap, kekLength, ukm)) → 对 CAEK 的 AES-Wrap → 以认证属性作为 AAD 对内容做 AEAD。它能运行，找到了正常流程，接收者侧的检查（版本、rid、KEM 和 KDF 标识、KDF 上下文、标签）都已就位。

有十五个目标有答案，全部用对照查询核查过。其中九个在仅具网络能力的攻击者面前得到完全证明：ML-KEM 私钥、成对 KEK 和 CAEK 的保密性（K01、K02.8、A05），内容保密性（K02），kemct、KDF 上下文和算法标识符与被接受密钥的绑定，以及 rid 的选择器角色（A08、A08.9），属性的篡改与重组（D01、D01.10），每条消息一个新鲜的 KEM 共享秘密（X01），同一 CAEK 下 nonce 不复用（X05），以及临时泄露被限制在一条消息内（X02，在 KEK 泄露攻击者下）。

六个目标的主张被证伪：无数据源认证（D01、D01.10、X03——接收方公钥的任何持有者都能让接收方接受其选定的内容），无前向保密（C01：即使发起方的临时密钥被擦除，静态私钥仍能解开每一条记录下来的消息），KEK 泄露会暴露该消息的 CAEK 和内容（C04），以及重放（X04）。除重放外，这些都在文稿本身中有陈述（RFC 9629 §7、RFC 9936 §4），作为已知结果的复现报告，而非发现。

唯一的发现是重放：内容类型中没有任何东西携带新鲜性，接收方也不保留记录，因此记录下来的 AuthEnvelopedData 会被再次接受；值为低，修改消息定义无法消除它，只有接收方侧状态或 profile 必须定义的新鲜性值才行（已说明为未检查）。每个答案都有一个限制：KEM、HKDF、密钥封装和 AEAD 都是理想的，因此对 ML-KEM-768 的计算强度不做任何主张；本协议中也不存在计数器或被替换的密钥，因此不需要有界模型。