# ADRF 模型仓储的安全性分析：存储确认与检索

本报告的其他语言版本：English。

19.7.0 · ProVerif+ Agent 报告

| | |
|---|---|
| **文稿** | 5G System; Analytics Data Repository Services; Stage 3 （协议定义） |
| **检查的性质** | 10 条：6 条成立，4 条不成立（4 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

Analytics Data Repository Function 提供 ML 模型管理服务：一个 NF 服务消费者（通常是包含模型训练逻辑的 NWDAF）把 ML 模型存储到仓储中，之后由 LMF 等消费者检索这些模型。存储请求携带模型标识、模型文件的地址，以及可选的允许检索该模型的消费者列表；仓储返回所创建的记录和一个存储事务标识。检索请求给出存储事务标识或模型标识，仓储返回记录，即模型标识和模型文件的地址。

存储请求结束时，消费者确信它所看到的记录和收到的存储事务标识都属于自己。检索请求结束时，消费者确信返回的条目对应它所请求的模型标识，并且仓储已核实它要么是存储该模型的消费者，要么是存储该模型的 NWDAF 所列出的消费者。对于每个请求，仓储确信发送该请求的消费者的身份，因为访问由 OAuth2 令牌授权。

涉及的各方：

- **包含 MTLF 的 NWDAF**：存储模型的 NF 服务消费者。持有：nfInstanceId、OAuth2 访问令牌。
- **ADRF**：ML 模型存储。持有：SBI 凭据。
- **LMF**：检索模型的 NF 服务消费者。持有：nfInstanceId、OAuth2 访问令牌。

图 1 展示了文稿所规定的交互流程。

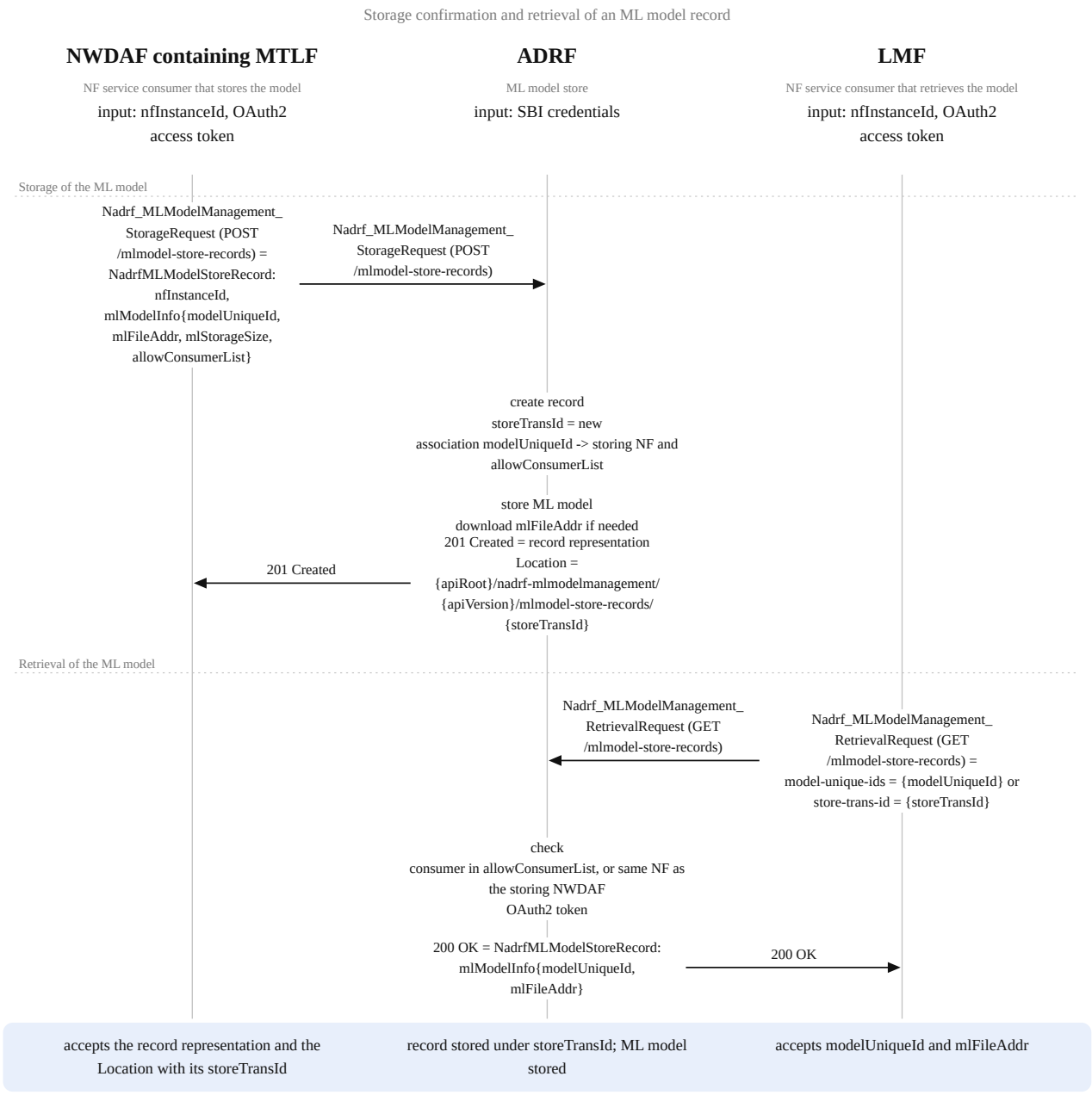



图 1. 按规范规定的协议。

分析的范围见附录 D：分析了什么、省略了什么，以及所依据的文稿章节。

## 2 发现

本次分析有 1 项发现：以下内容在文稿中并未说明。

在分析范围内，存储确认和检索检查成立：201 应答的是消费者自身的请求，仓储只把记录返回给存储该模型的 NWDAF，或返回给存储该模型的 NWDAF 所列出且持有有效令牌的消费者。有一项弱点是新发现的：任何已注册的 NF 消费者都可以在现有模型标识下注册第二条记录，仓储随后可能把该记录的文件地址返回给一个已获授权的消费者，于是该消费者下载的是攻击者选定的文件。文稿既未说明这一点，也未禁止它。这些结果依赖于以下假设：3GPP 服务化接口经过认证并受完整性保护（文稿将这一点交由其他规范规定），以及对 403 条件按“已列出或存储 NF”来解读。除此之外，证明和攻击覆盖任意数量的会话。

攻击图中的列与图 1 相同。红色表示攻击者的行为。黄色标记的是某一方发送了消息，但规定的交互流程在该处并没有这条消息，以及攻击所破坏的结果。

### 2.1 已注册的消费者可以接管另一 NF 的模型标识，并使其文件地址被返回

| | |
|---|---|
| **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** | 高 |
| **涉及的性质** | X01, A08, A08.8, Z03 （见第 3 节） |
| **原文** | s0048, s0080, s0081, s0255 |
| **依据** | ProVerif attack query `a08_addr`; Figures 2, 3, 4 and 5 |

**发生了什么。** 一个持有服务范围令牌的 NF 服务消费者存储了一条记录，该记录携带 NWDAF 已注册的 modelUniqueId，以及它自己的 mlFileAddr 和一个 allowConsumerList，该列表列出了它想要触达的消费者。ADRF 为每个存储请求保留一条记录，因此现在两条记录携带相同的标识。当 LMF 按该 modelUniqueId 检索时，ADRF 接受其请求，因为为该标识注册的允许列表匹配，并返回那条恶意记录，于是 LMF 收到的是恶意文件地址，而不是 NWDAF 的地址。

**攻击者需要什么。** 一个已注册的 NF 实例，以及一个针对单一服务范围 nadrf-mlmodelmanagement 的 OAuth2 令牌，s0295 已为整个服务给出该范围。不需要拦截，也不需要窃取凭据；恶意消费者自行选择其请求中的模型标识、文件地址和允许列表。

**攻击者能得到什么。** 按标识检索模型的消费者可能被交给一个攻击者选定的文件地址，从而下载恶意模型，替代 NWDAF 训练的模型，而它所看到的标识仍是预期的那个。

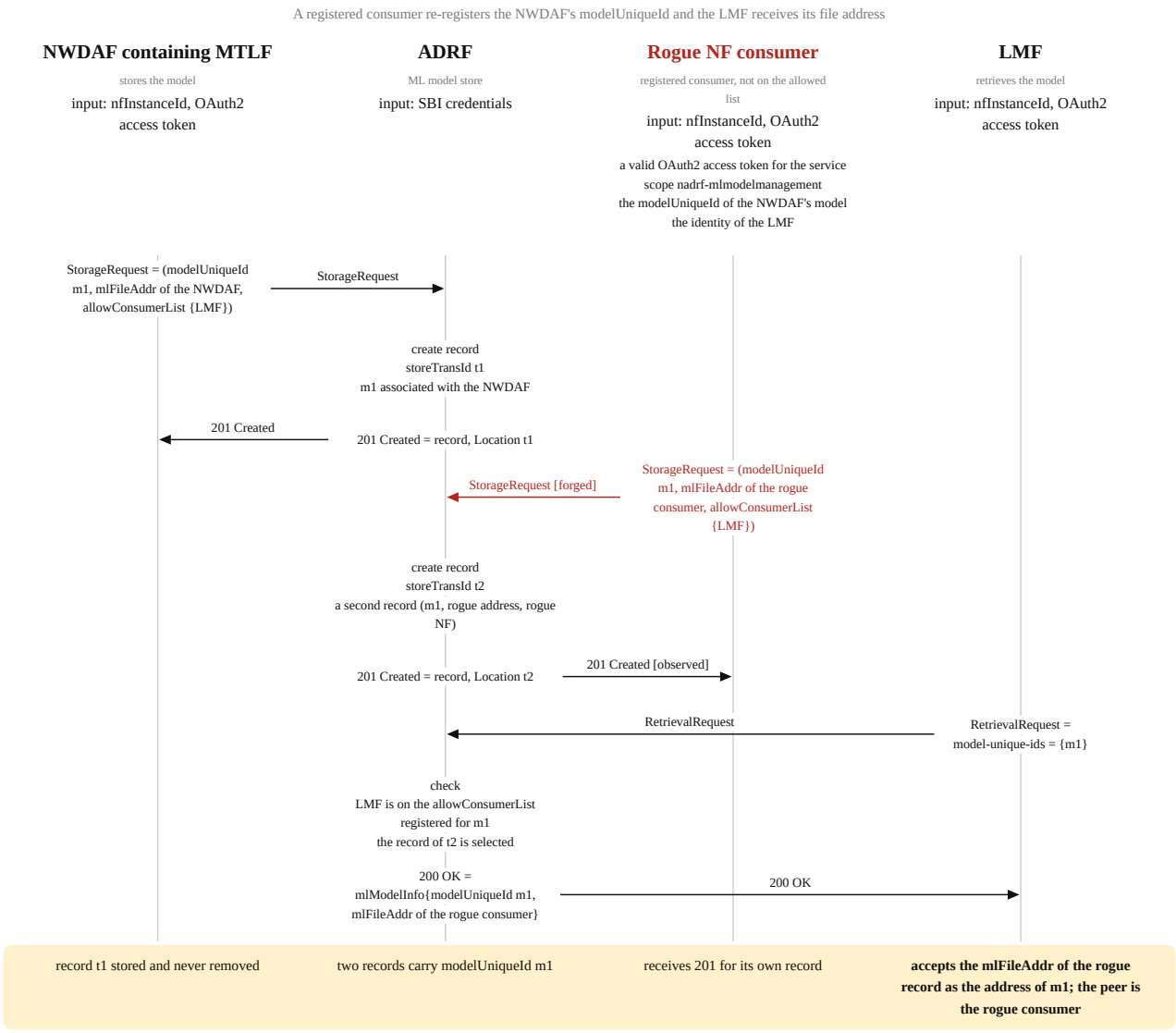



图 2. 已注册的消费者重新注册 NWDAF 的 modelUniqueId，LMF 收到它的文件地址。性质 X01；攻击查询 a08_addr。

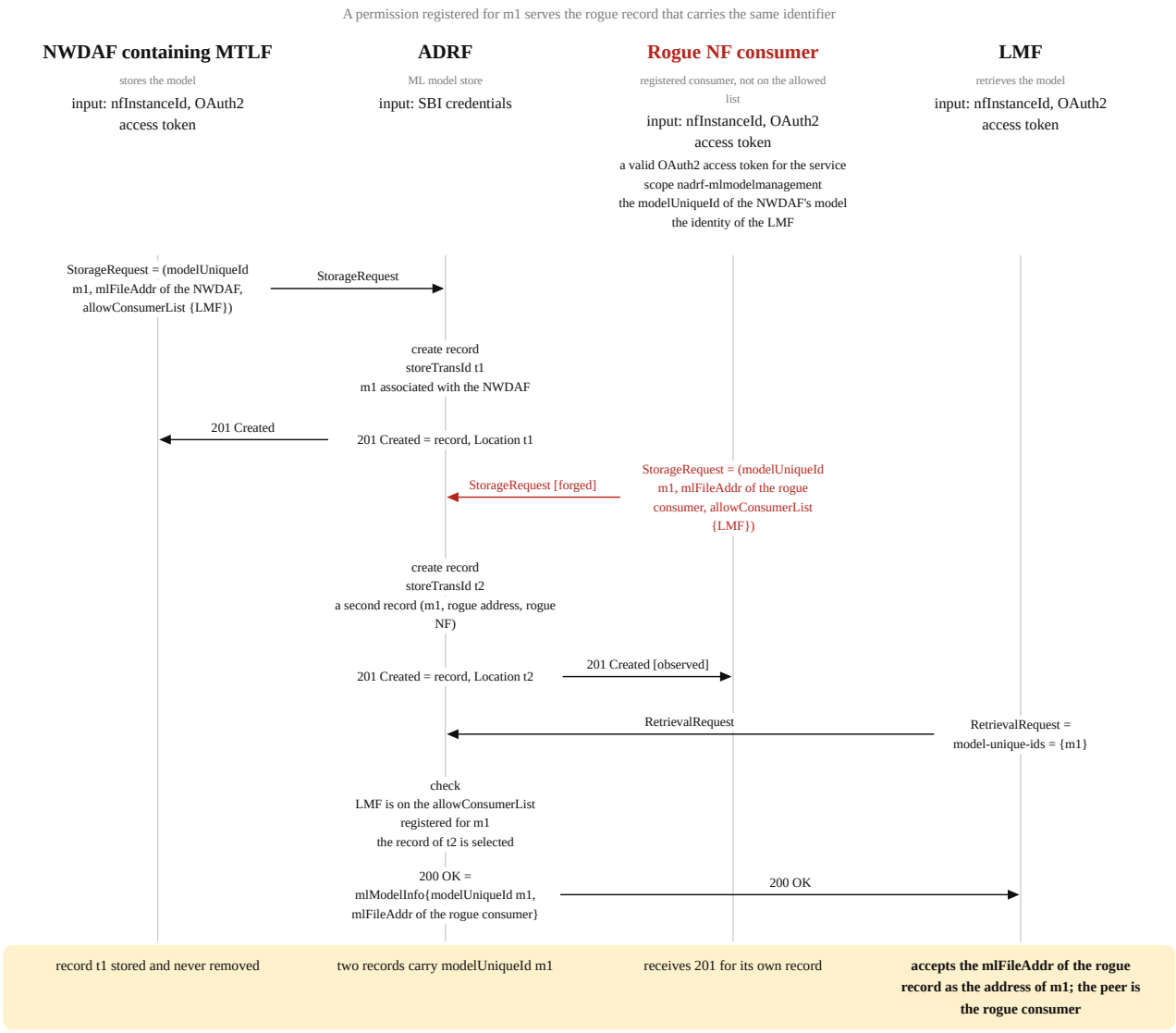



图 3. 为 m1 注册的一项许可服务于携带相同标识的恶意记录。性质 Z03；攻击查询 a08_addr。

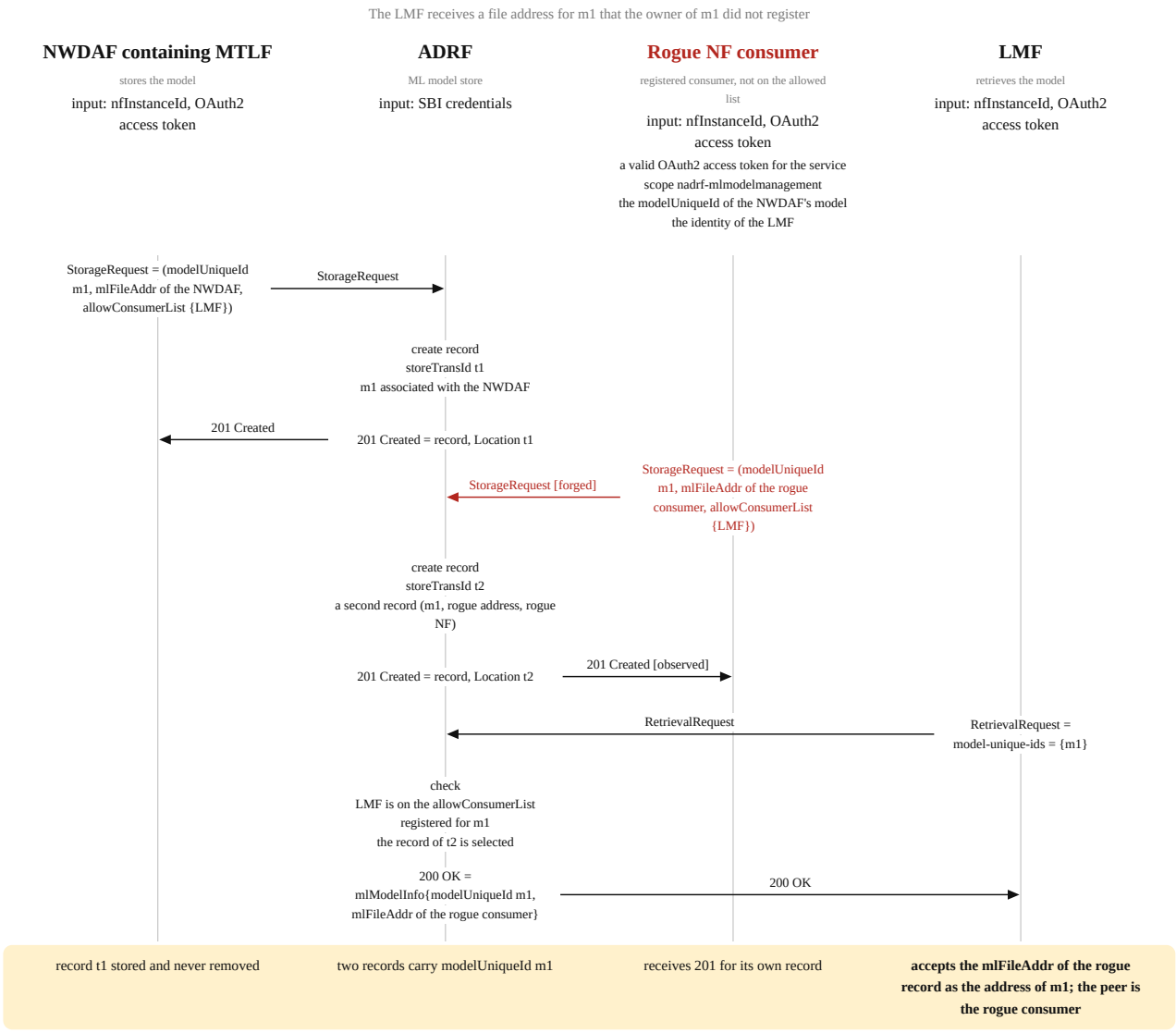



图 4. LMF 收到的是 m1 的一个文件地址，而该地址并非 m1 的所有者所注册。性质 A08；攻击查询 a08_addr。

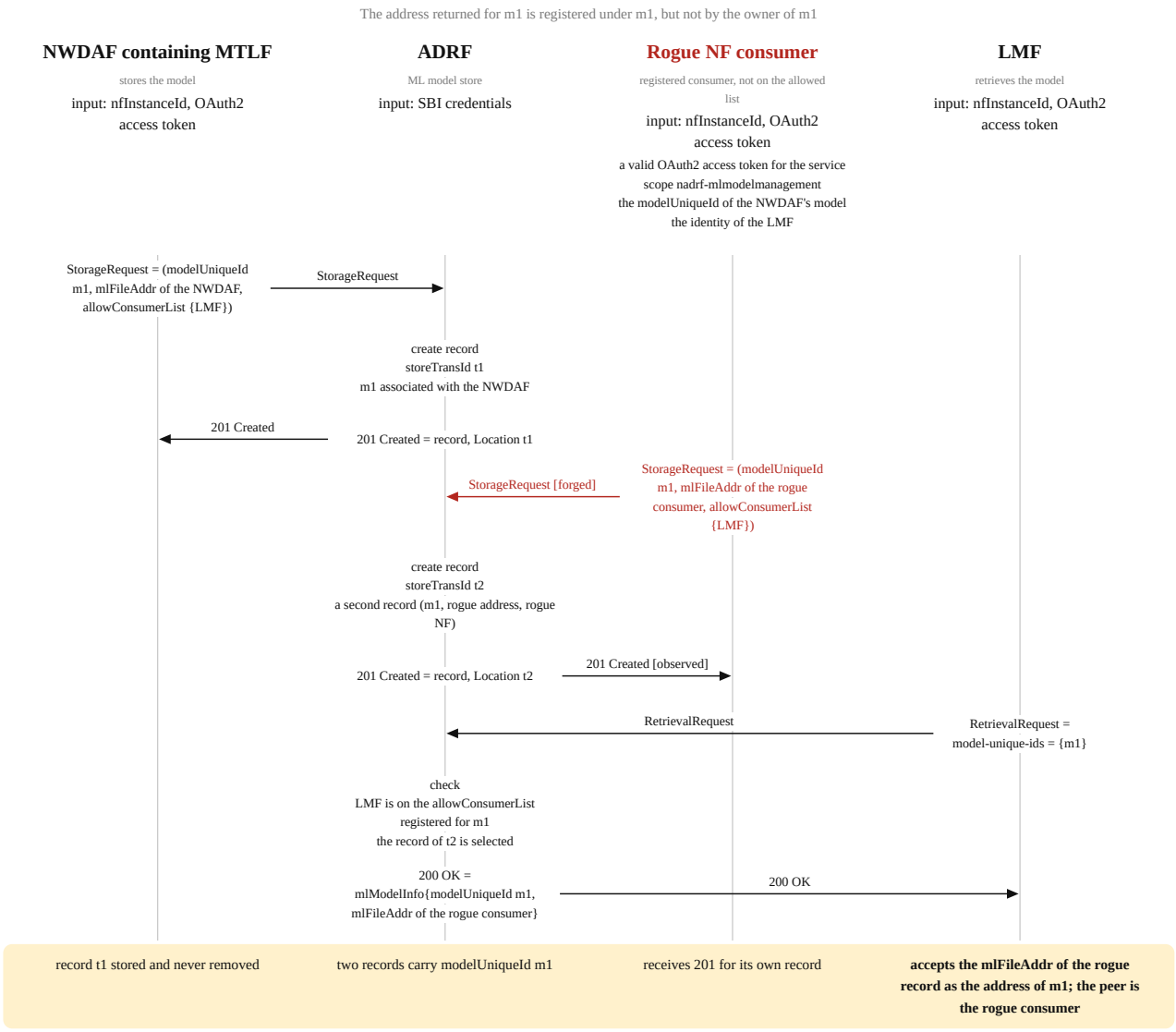



图 5. 为 m1 返回的地址注册在 m1 之下，但并非由 m1 的所有者注册。性质 A08.8；攻击查询 a08_addr。

**拟议修改。** ADRF 应当为每个 modelUniqueId 只保留一个关联（TS 23.288 第 5102 行已经这样要求），并对 modelUniqueId 已与另一不同 NF 实例关联的存储请求以 409 Conflict 拒绝，而不是为其创建第二条记录；此后按 modelUniqueId 检索时返回该唯一关联的记录。对已有关联的检查和创建新关联必须是同一个原子步骤。

**用 ProVerif 复核。** 未检查：这一失败取决于 ADRF 先记录两个存储请求中的哪一个：只拒绝重复标识的检查只会把攻击转移到恶意请求先被记录的那次运行，因为协议并不规定它们的顺序。要消除该问题，需要在分配时就把标识绑定到单个 NF，而消费者自行提供 modelUniqueId（s0255），因此这一改动超出了本模型所表示的消息流。

## 3 检查过的性质

每一行都是一条提交给 ProVerif 的性质。结果是在分析者对协议的模型上 ProVerif 给出的答案；价值以及文稿是否已经写明该结果，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 |
|---|---|---|---|---|---|
| Z01 | **仅授权消费者可检索。** ADRF 返回 200 OK 并给出模型记录（含 modelUniqueId 与 mlFileAddr）时，该请求者必须在该模型的允许消费者列表内，或与存储该模型的 NWDAF(含 MTLF)为同一 NF，且持有覆盖该操作的授权。 | 无 | 成立 | 中 | 是 |
| Z03 | **授权绑定到模型与请求者。** 一次检索所用的授权必须绑定到被请求的模型标识（modelUniqueId 或 storeTransId 所指记录）、请求者身份与检索动作；不得把对某模型或某消费者的许可用于另一模型或另一消费者。 | 无 | 不成立（1 个攻击） | 高 | 不确定 |
| A03 | **201 应答消费者自身的存储请求。** 消费者收到 201 Created 时，其返回的记录表示与 Location 中的 storeTransId 对应它自己发起的存储请求；消费者收到 200 OK 时，返回记录的 modelUniqueId 与 mlFileAddr 对应它请求的 store-trans-id 或 model-unique-ids。 | 无 | 成立 | 中 | 是 |
| A08 | **检索到的记录应答请求。** 检索响应中的每个模型条目都对应请求中给出的 store-trans-id 或 model-unique-ids 之一，且 mlFileAddr 是该 modelUniqueId 在存储时登记的地址；创建响应的 Location 指向本次创建的记录。 | 无 | 不成立（1 个攻击） | 高 | 否 |
| D01 | **消息在传输中不可被替换。** ADRF 接受的存储请求内容与消费者真实发送的内容一致；消费者接受的检索响应内容与 ADRF 真实返回的内容一致，未被中间人替换。 | 无 | 成立 | 低 | 不确定 |
| K01 | **访问令牌保持保密。** 攻击者不能获得消费者的 OAuth2 访问令牌或用于取得令牌的客户端凭据。 | 无 | 成立 | 低 | 不确定 |
| A03.7 | **存储确认绑定请求者与 storeTransId。** 消费者收到 201 Created 时，返回记录与 Location 中的 storeTransId 对应它自己发起的存储请求，且记录中的 nfInstanceId/nfSetId 为该请求者。 | 无 | 成立 | 低 | 是 |
| A08.8 | **检索条目与请求的标识符匹配。** 检索响应中的每个模型条目对应请求给出的 store-trans-id 或 model-unique-ids 之一，且 mlFileAddr 是该 modelUniqueId 在存储时登记的地址。 | 无 | 不成立（1 个攻击） | 高 | 否 |
| Z01.9 | **未列出的消费者得不到已存储记录。** 在固定的允许消费者列表与存储者身份下，不在列表内且与存储 NWDAF 不同的消费者不能获得 200 OK 的模型记录。 | 无 | 成立 | 中 | 是 |

| ID | 性质 | 假定的泄露 | 结果 | 价值 | 文稿是否已写 |
|---|---|---|---|---|---|
| X01 | **模型身份绑定到其存储者。** 当消费者按 modelUniqueId 检索时，它收到的 mlFileAddr 是拥有该 modelUniqueId 的 NF 所注册的地址；另一 NF 服务消费者不能注册一条携带相同 modelUniqueId 的第二条记录，并让其 mlFileAddr 随后被返回给消费者。 | 一个针对单一服务范围 nadrf-mlmodelmanagement 的有效 OAuth2 访问令牌，即一个已注册的 NF 服务消费者，且它不在其想要攻击的模型的 allowConsumerList 内；它可以发送自己的存储和检索请求，并可以选择自己存储请求中的 modelUniqueId、mlFileAddr 和 allowConsumerList。 | 不成立（1 个攻击） | 高 | 否 |

文稿写明、但起草清单中没有、由分析者补充的目标（见附录 D）：X01。

所提出的问题及分析者的回答：

| 问题 | 回答 | 依据 |
|---|---|---|
| Q1: 存储确认绑定请求者及 storeTransId | 是：消费者接受的 201 Created，及其 Location 中的 storeTransId，始终应答该消费者自己的存储请求，且记录以存储的 NF 实例作为其 nfInstanceId。 | A03, A03.7 |
| Q2: 检索结果绑定请求的 modelUniqueId/事务与返回地址 | 仅部分：返回的条目始终对应所请求的 modelUniqueId，但其 mlFileAddr 不一定是该模型所有者登记的地址，因为第二条记录可以携带同一标识符，ADRF 可能以它作答。 | A08, A08.8, X01 |
| Q3: 对固定授权快照检查未获准消费者是否可能取得记录 | 对于与其无关联的记录，未列入的消费者会被拒绝；但一个不在列表中且未存储该模型的消费者，仍可通过自行注册一条携带该 modelUniqueId 的记录来取得它，因此该保护并不延伸到标识符。 | Z01.9, Z01, X01 |

# 4 本次分析的局限

这里的结果是协议模型上的结果：它说明规定的消息与检查给出了什么保证、或者未能给出什么保证，密码学被视为理想，与实现无关。

正如分析者所述，本分析不覆盖以下内容：

- 3GPP 服务化接口被表示为每个已注册 NF 一条私有的认证通道。消息完整性、发送方认证和 OAuth2 令牌是假设的，而非推导得出的；文稿将其委托给 TS 29.500 和 TS 33.501。
- 只对按 modelUniqueId 检索进行了建模。按 store-trans-id 检索，以及一条记录持有多个 ML 模型、每个模型各有自己的允许消费者列表的情况，均未表示。
- s0083 中 403 的措辞（'and/or'）被建模为：当消费者在允许列表中，或是存储的 NWDAF 时允许。AND 解读（会拒绝已列入的消费者）未作分析。
- 建模了一个恶意消费者和一个检索消费者；记录存储是单调表，因此删除、更新和权限撤销不在模型范围内，正如范围所述。
- 存储或检索请求的部分失败以及 EnModelMgmt 特性（包括 modelStoreResult）未建模。
- NRF 的令牌签发、其范围、生存期和撤销不在模型范围内；令牌是认证 NF 实例的凭证。
- 在模型中 ADRF 是诚实的：不分析不诚实的存储库，也无法针对它陈述其行为的任何性质。

每条性质仍待解决的问题，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，而非证明器的回答：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，以及它有什么价值。

### Z01: 仅授权消费者可检索

当 ADRF 返回 200 OK 并给出模型记录（含 modelUniqueId 与 mlFileAddr）时，该请求者必须在该模型的允许消费者列表内，或与存储该模型的 NWDAF(含 MTLF) 为同一 NF，且持有覆盖该操作的授权。

ADRF 只将存储的记录连同模型文件地址交给存储该模型的 NWDAF，或交给存储该 NWDAF 列入该模型允许消费者列表的消费者，且只交给出示了有效访问令牌的请求者。两个条件都承载该性质：去掉允许列表检查或去掉令牌检查，都会使 ADRF 服务于错误的一方。对于完全没有令牌的消费者，拒绝是无条件的。

- 结果：成立；回答了所问：是；在附带注意事项下可用
- 价值中：s0083 和 s0286 要求对不被允许且非存储 NF 的消费者返回 403 RETRIEVAL_ML_MODEL_NOT_ALLOWED，s0292-s0295 要求该 API 使用 OAuth2 令牌；分析确认这些规则在实现时成立。
- 文稿中已陈述：是。文稿所述位置：main:1358、s0083、s0286、s0292、s0295
- 依据：3GPP 服务化接口被表示为每个已注册 NF 一条私有的、经过认证的通道，因此发送方身份和消息完整性是假设的而非推导得出的（TS 29.500、TS 33.501，在所选文本之外）；s0083 的 403 条件按以下解读建模：消费者在允许消费者列表中或是存储的 NWDAF 时即被允许；AND 解读会拒绝已列入的消费者，未建模；允许消费者列表按 modelUniqueId 保存；一条记录包含多个 ML 模型、每个模型各有列表的情况未建模；NRF 的令牌签发、范围和过期未建模；令牌是认证 NF 实例的凭证
- 待解决：s0083 中 and/or 措辞的策略解读；为单一服务范围签发的令牌能否由 NRF 限制到某个操作

### Z03: 授权绑定到模型与请求者

一次检索所用的授权必须绑定到被请求的模型标识（modelUniqueId 或 storeTransId 所指记录）、请求者身份与检索动作；不得把对某模型或某消费者的许可用于另一模型或另一消费者。

ADRF 据以决定一次检索的身份是发送该请求的身份，因此消费者不能以另一 NF 的身份请求某个模型，被窃取的令牌也使窃取者无法与被窃取的 NF 区分。主张的后半部分不成立：为某个 modelUniqueId 注册的许可被应用于携带该标识符的任意记录，因此为 NWDAF 的模型授予的许可被用来服务另一消费者在同一标识符下注册的记录。

- 结果：不成立（1 个攻击）；回答了所问：部分；在附带注意事项下可用
- 价值高：身份绑定是 s0083 和 s0292-s0295 所要求的，并在此得到确认；许可在两条携带相同 modelUniqueId 的记录之间被复用，这一点文稿中任何地方都未陈述，TS 23.288 第 5102 行只说 ADRF 维护模型标识符、存储 NF 与允许列表之间的关联。
- 文稿中已陈述：不明确。依据的原文：s0083、s0255、s0295
- 依据：发送方身份来自每 NF 通道（假设 TS 29.500/33.501 的保护）；模型中一条记录持有一个 ML 模型；多模型记录的按模型允许列表未表示；OAuth2 令牌的获取和内容未建模
- 待解决：是否要求 ADRF 拒绝针对已与另一 NF 实例关联的 modelUniqueId 的第二次存储请求

### A03: 201 应答消费者自身的存储请求

消费者收到 201 Created 时，其返回的记录表示与 Location 中的 storeTransId 对应它自己发起的存储请求；消费者收到 200 OK 时，返回记录的 modelUniqueId 与 mlFileAddr 对应它请求的 store-trans-id 或 model-unique-ids。

消费者接受的 201 Created 始终应答该消费者发送的存储请求，其携带的 storeTransId 被分配给该请求。另一事务的响应不会被接受，因为记录表示会对照所请求的模型进行检查。

- 结果：成立；回答了所问：是；在附带注意事项下可用
- 价值 中：s0049 和 s0165 要求 201 携带所创建的记录和含 storeTransId 的 Location，分析确认客户端侧绑定成立。
- 文稿中已陈述：是。文稿所述位置：s0042、s0049、s0165
- 依据：响应到达消费者自己的认证通道（假设 SBI 保护）；一条记录一个 ML 模型；记录表示被简化为其 modelUniqueId 和文件地址；无并发检查：对同一 modelUniqueId 的过期 201 会被接受用于对同一模型的后续请求
- 待解决：同一 NF 两次存储同一 modelUniqueId 且较早的 201 被重放时的行为

### A08: 检索到的记录应答请求

检索响应中的每个模型条目都对应请求中给出的 store-trans-id 或 model-unique-ids 之一，且 mlFileAddr 是该 modelUniqueId 在存储时登记的地址；创建响应的 Location 指向本次创建的记录。

消费者收到其所请求 modelUniqueId 的条目，因此响应不会在标识符之间混淆。它收到的地址不一定是该模型所有者登记的地址：如果另一 NF 消费者注册了携带同一 modelUniqueId 的第二条记录，ADRF 可能以该记录作答，消费者便下载了攻击者的文件。

- 结果：不成立（1 个攻击）；回答了所问：部分；在附带注意事项下可用
- 价值 高：检索了安全条款（s0291-s0295）、存储与检索条款（s0042-s0084）以及 TS 23.288 第 6.2B.5 和 6.2B.7 条，以寻找关于重复 modelUniqueId 的规则：文稿陈述对已存储的模型会创建新记录和新 storeTransId（s0048），且 ADRF 维护标识符与存储 NF 之间的关联（TS 23.288 第 5102 行），但从未说明第二个消费者不得注册同一标识符，也未说明此时检索返回哪条记录。
- 文稿中已陈述：否。依据的原文：s0048、s0080、s0081、s0255
- 依据：一条记录一个 ML 模型且仅按 modelUniqueId 检索；按 storeTransId 检索未建模；ADRF 被建模为返回一条匹配记录；文本未说明当多条记录携带同一标识符时返回哪一条；检查所用的允许消费者列表是为该 modelUniqueId 注册的那个
- 待解决：当多条记录携带所请求的 modelUniqueId 时，ADRF 返回哪一条

### D01: 消息在传输中不可被替换

ADRF 接受的存储请求内容与消费者真实发送的内容一致；消费者接受的检索响应内容与 ADRF 真实返回的内容一致，未被中间人替换。

路径上任何一方都不能替换模型标识符或文件地址：ADRF 只处理消费者发送的标识符和地址，消费者也只接受 ADRF 产生的响应。这一结果完全依赖一个假设：基于服务的接口经过认证并受完整性保护，而文稿将这一点交由 TS 29.500 和 TS 33.501 规定。

- 结果：Holds；回答了所问：是；可带保留意见使用
- 价值低：文稿只规定了消息内容（s0042、s0049、s0081），并将接口保护交由其他规范（s0292）；它们本身并未规定完整性性质。
- 文稿中是否明确说明：不明确。依据的原文：s0042、s0049、s0081、s0292

- 依赖：SBI 被建模为每个 NF 一条私有认证通道；没有这一假设，这些性质全都会轻易不成立，而任何失败都只是模型的产物，而非协议的产物；没有消息被建模为可跨连接重放
- 待定：TS 29.500 和 TS 33.501 所规定的 SBI 完整性

### K01: 访问令牌保持保密

攻击者不能获得消费者的 OAuth2 访问令牌或用于取得令牌的客户端凭据。

在所建模的交互中，NWDAF 的访问令牌从未暴露；网络上没有任何一方能获得它。这一结果说明不了太多，因为令牌只被用作认证消费者的凭据，而其由 NRF 签发的过程不在模型范围内。

- 结果：Holds；回答了所问：是；可带保留意见使用
- 价值低：s0292-s0293 说消费者在消费服务之前先从 NRF 取得令牌，但没有为其规定保密目标；令牌处理本身在 TS 29.510 和 TS 33.501 中。
- 文稿中是否明确说明：不明确。依据的原文：s0292、s0293
- 依赖：令牌由每个 NF 的私有凭据表示，从不发送；令牌格式、生命周期以及与 NRF 的客户端凭据交换均未建模
- 待定：NRF 交换中令牌的保护

### A03.7: 存储确认绑定请求者与 storeTransId

消费者收到 201 Created 时，返回记录与 Location 中的 storeTransId 对应它自己发起的存储请求，且记录中的 nfInstanceId/nfSetId 为该请求者。

ADRF 为已存储模型记录的 nfInstanceId 就是发送存储请求的 NF 实例，因此该记录及其 Location 属于请求者，而不属于第三方。

- 结果：Holds；回答了所问：是；可带保留意见使用
- 价值低：s0252 和 s0049 规定了记录及 201 中的返回；分析确认该字段由请求者填写。
- 文稿中是否明确说明：是。文稿中说明的位置：s0252、s0049、s0165
- 依赖：发送者身份来自每个 NF 的通道；记录被简化为存储 NF、modelUniqueId 和文件地址；nfSetId 及其他属性未表示
- 待定：nfSetId 作为唯一标识符（s0252 的 NOTE 1），此时在集合内无法区分存储实例

### A08.8: 检索条目与请求的标识符匹配

检索响应中的每个模型条目对应请求给出的 store-trans-id 或 model-unique-ids 之一，且 mlFileAddr 是该 modelUniqueId 在存储时登记的地址。

消费者接受的每个条目都是针对它请求的 modelUniqueId。该条目中的地址只保证是某个消费者在该标识符下登记的地址，而不保证是模型所有者登记的地址，因此主张的后半部分因与检索目标相同的原因而不成立。

- 结果：Fails（1 个攻击）；回答了所问：部分；可带保留意见使用
- 价值高：检索了存储、检索和安全条款以及 TS 23.288 6.2B.5/6.2B.7：文稿没有说明当多条记录带有请求的 modelUniqueId 时返回哪一条，也没有说明重复标识符会被拒绝。
- 文稿中是否明确说明：否。依据的原文：s0048、s0080、s0081
- 依赖：每条记录一个模型，按 modelUniqueId 检索，返回一条匹配记录；ADRF 保留为同一标识符创建的所有记录（s0048）
- 待定：重复 modelUniqueId 值的处理规则

### Z01.9: 未列出的消费者得不到已存储记录

在固定的允许消费者列表与存储者身份下，不在列表内且与存储 NWDAF 不同的消费者不能获得 200 OK 的模型记录。

对于与其无关的记录，既不在允许列表内也不是存储 NF 的消费者会被拒绝，模型中的恶意消费者只能收到它自己存储的记录。该主张对记录成立；若按 modelUniqueId 解读则不成立，因为恶意消费者可以获得带有另一个 NF 的 modelUniqueId 的记录，这正是模型身份目标下报告的攻击。

- 结果：Holds；回答了所问：是；可带保留意见使用
- 价值中：s0083 和 s0286 规定了对既不被允许也不是存储 NF 的消费者的拒绝；模型对记录确认了这一点。
- 文稿中是否明确说明：是。文稿中说明的位置：s0083、s0286
- 依赖：快照中每个 modelUniqueId 一条记录；重复情况是单独报告的攻击；允许消费者列表由存储消费者的请求固定
- 待定：按 modelUniqueId 的解读，由模型身份目标回答

### X01: 模型身份绑定到其存储者

当消费者按 modelUniqueId 检索时，它收到的 mlFileAddr 是该 modelUniqueId 所有者 NF 登记的地址；另一个 NF 服务消费者不能登记一条带有相同 modelUniqueId 的第二条记录，且其 mlFileAddr 随后被返回给消费者。

该主张不成立。一个持有该服务令牌的已注册 NF 消费者存储了第二条记录，该记录带有 NWDAF 的 modelUniqueId，但使用自己的文件地址，并带有列出 LMF 的允许消费者列表；ADRF 保留两条记录并提供恶意的那条，于是 LMF 下载了恶意文件，却以为它是 NWDAF 的模型。

- 结果：Fails（1 个攻击）；回答了所问：是；可带保留意见使用
- 价值高：所选文本的任何条款以及 TS 23.288 6.2B.5/6.2B.7 的任何条款都没有规定 modelUniqueId 绑定到首次登记它的 NF，也没有规定第二次登记会被拒绝；s0048 对记录的说法相反，TS 23.288 第 5102 行只说明 ADRF 维护标识符、存储 NF 和允许列表之间的关联。
- 文稿中是否明确说明：否。依据的原文：s0048、s0080、s0081、s0255
- 依赖：已注册的 NF 消费者可以为单一服务范围取得令牌，并在自己的请求中选择 modelUniqueId、文件地址和允许列表；每条记录一个模型；ADRF 返回匹配记录之一；SBI 本身被假定受保护；该攻击不使用拦截
- 待定：运营商能否防止 modelUniqueId 值被其他 NF 猜测或选择

# 附录 C：主张和证明器的答案

证明器为每个目标记录的：达到的级别、达到该级别的查询、是否检查了对照，以及各项攻击及分析者对每项攻击的判定。

| 目标 | 级别 | 界 | 查询 | 对照 | 攻击 | 主张 |
|---|---|---|---|---|---|---|
| Z01 | proved | | z01_auth | checked | | 3 proved |
| Z03 | proved | | z03_bind | checked | a08_addr (holds_in_source) | 1 proved, 1 attack |
| A03 | proved | | a03_bind | checked | | 1 proved |
| A08 | proved | | a08_req | checked | a08_addr (holds_in_source) | 1 proved, 1 attack |
| D01 | proved | | d01_get | checked | | 2 proved |
| K01 | proved | | k01_tok | checked | | 1 proved |
| A03.7 | proved | | a037_nf | checked | | 1 proved |
| A08.8 | proved | | a08_req | checked | a08_addr (holds_in_source) | 1 proved, 1 attack |
| Z01.9 | proved | | z019_auth | checked | | 1 proved |
| X01 | none | | | | a08_addr (holds_in_source) | 1 attack |

逐条主张，按分析者对每条主张的措辞；每条主张的答案来自证明器：

| 目标 | 主张 | 攻击者 | 回答 | 查询 | 对照 |
|---|---|---|---|---|---|
| Z01 | ADRF 仅向该 NWDAF 或模型 allowConsumerList 上的消费者发送携带存储 NWDAF 所注册 mlFileAddr 的 200 OK 记录。 | `stolen_token` | proved | z01_auth | checked |
| Z01 | 200 OK 仅发送给为其身份出示了有效访问令牌的请求者。 | `stolen_token` | proved | z01_tok | checked |
| Z01 | 完全不持有访问令牌的请求者永远不会收到 ADRF 发送的记录。 | `stolen_token` | proved | z01_anon | checked |
| Z03 | ADRF 据以裁决检索的身份就是发送请求的身份；消费者不能以另一 NF 的身份提出请求。 | `stolen_token` | proved | z03_bind | checked |
| Z03 | 针对某个 modelUniqueId 持有的权限不会被用来服务于携带该标识符的另一条记录：消费者收到的是该模型所有者注册的 mlFileAddr，而不是另一 NF 为同一 modelUniqueId 注册的地址。 | `rogue_consumer` | attack (holds_in_source) | a08_addr | |
| A03 | 消费者接受的 201 Created 与其发送的存储请求相对应：记录表示和 Location storeTransId 回答的是它自己的 POST。 | `rogue_consumer` | proved | a03_bind | checked |
| A08 | 消费者针对某个 modelUniqueId 收到的 mlFileAddr 是该模型所有者在存储时注册的地址。 | `rogue_consumer` | attack (holds_in_source) | a08_addr | |
| A08 | 消费者接受的每个模型条目都对应其请求的 modelUniqueId。 | `rogue_consumer` | proved | a08_req | checked |
| D01 | 到达 ADRF 的检索请求中的 modelUniqueId 就是消费者发送的那个。 | `rogue_consumer` | proved | d01_get | checked |
| D01 | 到达 ADRF 的存储请求的内容就是消费者发送的内容。 | `rogue_consumer` | proved | d01_store | checked |
| K01 | 攻击者无法获得 NF 服务消费者的访问令牌或客户端凭证。 | `network` | proved | k01_tok | checked |

| 目标 | 主张 | 攻击者 | 回答 | 查询 | 对照 |
|---|---|---|---|---|---|
| A03.7 | ADRF 创建的记录中的 nfInstanceId/nfSetId 是发送存储请求的 NF 实例。 | `rogue_consumer` | proved | a037_nf | checked |
| A08.8 | 针对所请求的 modelUniqueId 返回的 mlFileAddr 是该模型所有者在存储时注册的地址。 | `rogue_consumer` | attack (holds_in_source) | a08_addr | |
| A08.8 | 消费者接受的每个模型条目都对应其请求的 modelUniqueId。 | `rogue_consumer` | proved | a08_req | checked |
| Z01.9 | 既不在所存储模型的 allowConsumerList 中、也不是存储该模型的 NF 的消费者，不会收到该模型的 200 OK 记录。 | `rogue_consumer` | proved | z019_auth | checked |
| X01 | 检索某个 modelUniqueId 的消费者收到的是该模型所有者所属 NF 注册的 mlFileAddr；其他任何 NF 都不能针对该标识符返回自己的地址。 | `rogue_consumer` | attack (holds_in_source) | a08_addr | |

# 附录 D：分析是怎么跑的

LLM 分析者根据文稿构建协议模型，并将每条性质交给 ProVerif；第 3 节中的每个结果都是证明器给出的回答。

ProVerif+ Agent 会话 session-20261114T183856，2026-11-14。状态：已完成。模型：deepseek-v4-1-flash-261017。步数：50，证明器 7 秒，实际耗时 15 分钟，费用 0.104 USD。

ProVerif：运行 128 次证明，128 次判定，0 次无答案，5 秒，记录 42 个回答。

## 分析了什么

| | |
|---|---|
| **协议** | ADRF 模型仓储：存储确认与检索 |
| **版本** | 19.7.0 |
| **分析范围** | 固定一个存储事务及不可变模型记录，模型已在本地/已完成下载；检查创建确认与按 storeTransId/modelUniqueId 检索。更新、删除与恶意文件下载放到后续。 |
| **不在范围内** | 403 条件存在 and/or 措辞，需要与 Stage 2/数据类型对照并记录采用的策略解释；返回下载地址的保密不等同于模型文件保密；持续更新、删除、撤销权限不是首轮符号模型的结论。仅证明明确模型及攻击者条件下的性质，不能把缺失检查、完整安全信道或全诚实 NF 当作待证结论的替代。 |
| **原文** | 295 个带编号的原文单元（54906 字节），来自下列章节。本报告中诸如 s0001 的引用即其中一个原文单元 |

分析所依据的文稿章节：

| 文稿 | 章节 | 标题 |
|---|---|---|
| main | 4.3 | Nadrf_ MLModelManagement Service |
| main | 4.3.1 | Service Description |
| main | 4.3.1.1 | Overview |
| main | 4.3.1.2 | Service Architecture |
| main | 4.3.1.3 | Network Functions |
| main | 4.3.1.3.1 | Analytics Data Repository Function (ADRF) |
| main | 4.3.1.3.2 | NF Service Consumers |
| main | 4.3.2 | Service Operations |
| main | 4.3.2.1 | Introduction |
| main | 4.3.2.2 | Nadrf_MLModelManagement_StorageRequest service operation |
| main | 4.3.2.2.1 | General |
| main | 4.3.2.2.2 | Request Storage of ML model(s) |
| main | 4.3.2.2.3 | Update Storage of ML model(s) |
| main | 4.3.2.3 | Nadrf_MLModelManagement_RetrievalRequest service operation |
| main | 4.3.2.3.1 | General |
| main | 4.3.2.3.2 | Request and get stored ML model(s) from ADRF ML Model Store |
| main | 4.3.2.4 | Nadrf_MLModelManagement_Delete service operation |

| 文稿 | 章节 | 标题 |
|---|---|---|
| main | 4.3.2.4.1 | General |
| main | 4.3.2.4.2 | Requesting removal of stored ML model(s) |
| main | 4.3.2.4.3 | Requesting removal of stored ML model(s) using unique ML model identifier |
| main | 5.2 | Nadrf_MLModelManagement Service API |
| main | 5.2.1 | Introduction |
| main | 5.2.2 | Usage of HTTP |
| main | 5.2.2.1 | General |
| main | 5.2.2.2 | HTTP standard headers |
| main | 5.2.2.2.1 | General |
| main | 5.2.2.2.2 | Content type |
| main | 5.2.2.3 | HTTP custom headers |
| main | 5.2.3 | Resources |
| main | 5.2.3.1 | Overview |
| main | 5.2.3.2 | Resource: ADRF ML Model Store Records |
| main | 5.2.3.2.1 | Description |
| main | 5.2.3.2.2 | Resource Definition |
| main | 5.2.3.2.3 | Resource Standard Methods |
| main | 5.2.3.2.3.1 | POST |
| main | 5.2.3.2.3.2 | GET |
| main | 5.2.3.2.4 | Resource Custom Operations |
| main | 5.2.3.3 | Resource: Individual ADRF ML Model Store Record |
| main | 5.2.3.3.1 | Description |
| main | 5.2.3.3.2 | Resource Definition |
| main | 5.2.3.3.3 | Resource Standard Methods |
| main | 5.2.3.3.3.1 | DELETE |
| main | 5.2.3.3.3.2 | PUT |
| main | 5.2.3.3.4 | Resource Custom Operations |
| main | 5.2.4 | Custom Operations without associated resources |
| main | 5.2.4.1 | Overview |
| main | 5.2.4.4 | Operation: remove-stored-mlmodel |
| main | 5.2.4.4.1 | Description |
| main | 5.2.4.4.2 | Operation Definition |
| main | 5.2.5 | Notifications |
| main | 5.2.6 | Data Model |
| main | 5.2.6.1 | General |
| main | 5.2.6.2 | Structured data types |
| main | 5.2.6.2.1 | Introduction |

| 文稿 | 章节 | 标题 |
|---|---|---|
| main | 5.2.6.2.2 | Type: NadrfMLModelStoreRecord |
| main | 5.2.6.2.3 | Type: MLModelInfo |
| main | 5.2.6.2.4 | Type: MLModel |
| main | 5.2.6.2.5 | Type: MLModelDelResult |
| main | 5.2.6.2.6 | Type: AllowedConsumer |
| main | 5.2.6.2.7 | Type: ModelStoreResult |
| main | 5.2.6.3 | Simple data types and enumerations |
| main | 5.2.6.3.2 | Enumeration: StoreResult |
| main | 5.2.6.4 | Data types describing alternative data types or combinations of data types |
| main | 5.2.7 | Error Handling |
| main | 5.2.7.1 | General |
| main | 5.2.7.2 | Protocol Errors |
| main | 5.2.7.3 | Application Errors |
| main | 5.2.8 | Feature negotiation |
| main | 5.2.9 | Security |

原文步骤在文稿中定位到的协议目的：固定一个存储事务及不可变模型记录，模型已在本地/已完成下载；检查创建确认与按 storeTransId/modelUniqueId 检索。更新、删除与恶意文件下载放到后续。

## 对照文稿定性质

在编写查询之前，分析者将草拟的性质列表与文稿本身所述的协议目的进行对照。名称、性质所涉及的内容以及文稿所述内容均为分析者的解读；所提及的位置在文稿中确实存在。

| ID | 简称 | 对象 | 文稿怎么说 | 出处 |
|---|---|---|---|---|
| Z01 | 仅授权消费者可检索 | ADRF 只向满足以下条件的 NF 服务消费者返回 NadrfMLModelStoreRecord（modelUniqueId、mlFileAddr）：该消费者在该 ML 模型的 allowConsumerList 中，或者是存储该模型的、包含 MTLF 的 NWDAF 的 NF，并且持有该操作的访问令牌。 | s0083 要求：当消费者不在允许的 NF 消费者列表中，和/或不是存储该模型的 NF 时，返回 403 RETRIEVAL_ML_MODEL_NOT_ALLOWED；s0286 给出了该错误的名称；s0292-s0293 要求该 API 使用 OAuth2 访问令牌；s0295 为整个服务只给出一个 scope。 | s0083, s0286, s0292, s0293, s0295 |
| Z03 | 授权绑定到模型与请求者 | 检索时检查的权限是：实际被请求的模型标识符所对应的权限，由实际发送请求的身份持有，且仅针对检索操作；ADRF 不会把一个模型或一个消费者的权限用于另一个模型或另一个消费者。 | GET 携带 store-trans-id 或 model-unique-ids（s0078、s0169），403 测试按每个 ML 模型进行（s0083），allowConsumerList 是每个模型的属性（s0255）；s0295 只给出一个服务级 scope，因此令牌本身不绑定服务级以下的任何内容。 | s0078, s0083, s0255, s0264, s0295 |

| ID | 简称 | 对象 | 文稿怎么说 | 出处 |
|---|---|---|---|---|
| A03 | 201 应答消费者自身的存储请求 | 消费者接受的 201 Created（带有记录表示和 Location storeTransId）对应该消费者发送的存储请求。 | s0042 和 s0045：消费者 POST 一个 NadrfMLModelStoreRecord，ADRF 分配一个 storeTransId；s0049 和 s0165：201 携带记录表示和带 storeTransId 的 Location。 | s0042, s0045, s0049, s0165 |
| A08 | 检索到的记录应答请求 | 200 OK 中的每个模型条目对应请求中的 store-trans-id 或 model-unique-ids，其 mlFileAddr 是拥有该 modelUniqueId 的 NF 在存储时登记的地址。 | s0078：GET 通过 store-trans-id 或 model-unique-ids 标识记录；s0081：200 响应体携带 MLModelInfo，其中包含 modelUniqueId 和存储在 ADRF 中的 ML 模型文件的地址。 | s0078, s0081, s0169 |
| D01 | 消息在传输中不可被替换 | ADRF 处理的内容是消费者发送的内容，消费者接受的内容是 ADRF 返回的内容；没有其他方在传输中替换模型标识符或文件地址。 | s0042、s0049 和 s0081 描述了请求和响应体；SBI 本身的保护委托给 TS 29.500 和 TS 33.501（s0292），这些不在选定文本范围内。 | s0042, s0049, s0081, s0292 |
| K01 | 访问令牌保持保密 | NF 服务消费者的 OAuth2 访问令牌（客户端凭据）不会被攻击者获取。 | s0292-s0293：访问通过 OAuth2 客户端凭据授予方式授权，NRF 充当授权服务器；消费者在消费服务前获取令牌。 | s0292, s0293 |
| A03.7 | 存储确认绑定请求者与 storeTransId | 201 中返回的记录及其 Location storeTransId 属于该消费者的存储请求，记录中的 nfInstanceId/nfSetId 是该请求者。 | s0252：NadrfMLModelStoreRecord 携带包含 MTLF 的 NWDAF 的 nfInstanceId 或 nfSetId；s0049 和 s0165：201 返回记录和带 storeTransId 的 Location。 | s0252, s0049, s0165 |
| A08.8 | 检索条目与请求的标识符匹配 | 检索响应的每个模型条目对应请求中的一个 store-trans-id 或 model-unique-ids 值，其 mlFileAddr 是拥有该 modelUniqueId 的 NF 在存储时登记的地址。 | s0078、s0081 和 s0169 定义了请求标识符以及返回的 MLModelInfo（包含 modelUniqueId 和 mlFileAddr）。 | s0078, s0081, s0169 |
| Z01.9 | 未列出的消费者得不到已存储记录 | 在 allowConsumerList 固定且存储 NWDAF 固定的情况下，既不在列表中也不是存储者的消费者无法获得该模型的 200 OK 记录。 | s0083 和 s0286：对于不被允许且不是存储 NF 的消费者，返回 403 RETRIEVAL_ML_MODEL_NOT_ALLOWED。 | s0083, s0286 |
| X01 | 模型身份绑定到其存储者 | 当消费者检索一个被多个 NF 服务消费者存储过的 modelUniqueId 时，它会收到哪个 mlFileAddr。 | s0255 称 modelUniqueId 为唯一的 ML 模型标识符；s0080 要求 ADRF 根据请求参数找到 ML 模型；s0048 指出，对于已经存储的模型，ADRF 仍会创建新记录并分配新的 storeTransId。 | s0048, s0080, s0255 |

分析者依据文稿补充的性质：X01.

## 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 36 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 8 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 |
|---|---|---|
| K02 | 载荷与端到端保密性 | Z01 |

| 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 | 不适用 | 交换的内容中不涉及不可区分性受到威胁的秘密值；记录字段是标识符，令牌由 K01 覆盖。 |
| K04 | 密钥用途、方向与会话分离 | 不适用 | 该服务没有密钥层级、方向或会话派生。 |
| A01 | 对端曾参与性（Aliveness） | A03 | |
| A02 | 弱一致性认证 | A03 | |
| A04 | 单射一致性认证 | A03 | |
| A05 | 隐式密钥认证 | 不适用 | 该服务不建立会话密钥。 |
| A06 | 显式密钥确认 | 不适用 | 201 Created 确认资源创建，而不是密钥持有。 |
| A07 | 身份—凭据—密钥绑定 | Z03 | |
| R01 | 认证／挑战新鲜性 | A03 | |
| R02 | 消息级抗重放 | A03 | |
| R03 | 业务操作至多一次 | 不适用 | s0048 指出，对于已经存储的模型，ADRF 仍会创建新记录并分配新的 storeTransId，因此该服务不作出至多一次的声明。 |
| R04 | 消息接受顺序／窗口安全 | 不适用 | 该服务不定义序列号、窗口或排序规则。 |
| R05 | 抗降级／安全策略保持 | 不适用 | 唯一协商的特性（EnModelMgmt，s0290）会增加 allowConsumerList 和多结果；它不是可以被降低的安全级别。 |
| C01 | 前向保密性（FS） | 不适用 | 该服务不建立会话密钥。 |
| C02 | 抗密钥泄露冒充（KCI） | 不适用 | 该服务不建立会话密钥。 |
| C03 | 已知密钥安全／泄露隔离 | 不适用 | 没有多会话密钥派生或密钥层级。 |
| C04 | 临时秘密／部分状态泄露韧性 | 不适用 | 文稿未描述该服务的临时秘密或部分状态泄露。 |
| C05 | 泄露后安全恢复（PCS） | 不适用 | 该服务没有泄露后恢复机制。 |
| C06 | 注册凭据泄露后的增强认证 | 不适用 | 该服务不是 PAKE，也不持有此类注册记录。 |
| C07 | 抗离线字典攻击 | 不适用 | 不使用口令或其他低熵秘密；令牌从 NRF 获取（s0293）。 |
| C08 | 指定原语失效下的目标保持 | 不适用 | 文稿未说明原语失效下的目标；传输保护委托给 TS 29.500/33.501（s0292）。 |
| Z02 | 委托权限不扩张 | 不适用 | 该服务未描述可能扩大权限的委托链或代理。 |
| Z04 | 撤销与过期生效 | 不在范围内 | 任务说明明确将权限的删除和撤销留到后续轮次。 |
| Z05 | 跨租户／跨服务安全隔离 | Z01 | |
| S01 | 加密 nonce 唯一性 | 不适用 | 该服务不使用加密 nonce。 |
| S02 | 重传／密钥重安装安全 | 不适用 | 该服务不安装密钥，也不定义重密钥。 |
| S03 | 抗回滚与状态单调性 | 不适用 | 该服务不定义可能被回滚的版本或纪元单调性。 |
| S04 | 跨阶段安全上下文一致性 | 不适用 | 该服务只跨越单个存储和检索阶段，不跨阶段继承安全上下文。 |
| S05 | 终止与旧状态失效 | 不在范围内 | 任务说明将删除和撤销留到后续轮次。 |
| P01 | 身份匿名性 | 不适用 | 各方是具名的 NF 实例，其标识符携带在记录中；文稿未说明匿名性要求。 |
| P02 | 会话不可关联性 | 不适用 | 文稿未说明不可关联性要求，且该服务使用固定的 NF 实例标识符。 |
| P03 | 用户／凭据存在性隐藏 | 不适用 | 文稿未说明存在性隐藏要求；当记录不存在时，s0084 返回 204 No Content。 |

| 条目 | 目录里的性质 | 作为哪条性质检查 | 不检查的原因 |
|---|---|---|---|
| V01 | 条件性进展／错误恢复 | 证明器问不了 | 进展和错误恢复是活性性质，符号轨迹模型无法被询问这类性质。 |
| V02 | 无非预期死锁 | 证明器问不了 | 无死锁是活性性质，符号轨迹模型无法被询问这类性质。 |
| V03 | 受限资源滥用／抗放大 | 证明器问不了 | 资源消耗和放大超出了事件计数所能表达的范围。 |

## 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 17 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 8 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 | NF 服务消费者 | 参与方 | `rogue_consumer` | |
| S02 | ADRF | 参与方 | 按定义全部失守 | 不诚实的 ADRF 持有全部记录并自行应答检索；无法对其自身行为提出任何性质，而在该假设下文稿也未提出任何性质。 |
| S03 | NRF | 参与方 | 按定义全部失守 | 被攻陷的授权服务器可为任意 NF 身份签发令牌，因此这里所有基于身份的授权请求在定义上即告失效；文稿对此情形未作任何说明。 |
| S04 | 未获准的 NF 服务消费者 | 参与方 | `rogue_consumer` | |
| S05 | ML 模型存储记录 | 值 | `rogue_consumer` | |
| S06 | modelUniqueId | 值 | 攻击者得不到更多 | modelUniqueId 是由存储该记录的消费者选定、并在请求与响应中携带的标识符；知道它并不能带来网络攻击者所不具备的能力。 |
| S07 | mlFileAddr | 值 | `rogue_consumer` | |
| S08 | allowConsumerList | 值 | 攻击者得不到更多 | allowConsumerList 由存储该记录的消费者在其自己的请求中提供，并在记录中返回给获授权的消费者；持有它本身并不授权检索，这一点由 Z01 回答。 |
| S09 | OAuth2 访问令牌 | 值 | `stolen_token` | |
| S10 | storeTransId | 值 | 攻击者得不到更多 | storeTransId 返回给存储该记录的消费者，并出现在 Location 头中；它不是凭据，Z01 会独立于它校验请求方。 |
| S11 | 访问令牌 | 值 | `stolen_token` | |
| S12 | NadrfMLModelStoreRecord | 值 | `rogue_consumer` | |
| S13 | store-trans-id | 值 | 攻击者得不到更多 | store-trans-id 是用于指明某条记录的请求参数；知道它并不赋予检索权，这一点由 Z01 校验。 |
| S14 | model-unique-ids | 值 | 攻击者得不到更多 | model-unique-ids 是请求参数；知道某个 modelUniqueId 并不赋予检索权，这一点由 Z01 校验。 |
| S15 | Location | 值 | 攻击者得不到更多 | Location 的值是交给创建该记录的消费者的记录 URI；它不携带任何授权。 |
| S16 | 客户端凭据 | 值 | `stolen_token` | |
| S17 | NadrfMLModelStoreRecord.nfInstanceId/nfSetId | 值 | `rogue_consumer` | |

## 每条性质在各攻击者条件下的答案

攻击者由分析者在编写查询之前设定；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”是为该性质设定的、证明器对其留下未决主张或未记录任何内容的攻击者集合；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、修改和发送消息，不持有任何一方的秘密 | | |
| `rogue_consumer` | 一个针对单一服务范围 nadrf-mlmodelmanagement 的有效 OAuth2 访问令牌，即一个已注册的 NF 服务消费者，但它不在其想访问的模型的 allowConsumerList 中；它能发送自己的存储与检索请求，并能选择自己存储请求中的 modelUniqueId、mlFileAddr 和 allowConsumerList。 | s0295 为整个服务定义了单一范围，在资源或操作层面没有附加范围，因此每个获得令牌的消费者都可以调用每个操作；s0083 仅通过 allowConsumerList 和存储的 NF 限制检索；s0252/s0255 允许存储的消费者提供 modelUniqueId、mlFileAddr 和 allowConsumerList。 | s0252, s0255, s0083, s0295 |
| `stolen_token` | 存储该模型的 NF 服务消费者（含 MTLF 的 NWDAF）的 OAuth2 访问令牌和客户端凭据，在交换之前获得；它能通过自己的连接向 ADRF 出示该令牌。 | s0292-s0293：ADRF 通过 NRF 签发的 OAuth2 持有者令牌授权访问，因此持有该令牌者即以该消费者身份获得授权；文稿除客户端凭据授权外未描述任何令牌绑定。 | s0292, s0293 |

| ID | `network` | `rogue_consumer` | `stolen_token` |
|---|---|---|---|
| Z01 | 成立 | 成立 | 成立 |
| Z03 | 成立 | 不成立 | 成立 |
| A03 | 成立 | 成立 | |
| A08 | 成立 | 不成立 | |
| D01 | 成立 | 成立 | |
| K01 | 成立 | | |
| A03.7 | 成立 | 成立 | |
| A08.8 | 成立 | 不成立 | |
| Z01.9 | 成立 | 成立 | |
| X01 | | 不成立 | |

## 原文和目标

原文：295 个原文单元，54906 字节，冻结于 source-delivery-01.json。 目标：6 个已批准（Z01、Z03、A03、A08、D01、K01）；所提问题：3 个。 目标清单在事前按草稿获批，使用 --approve-targets。

宿主按其记录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-4d0742600750ac68a8578cc80b88

本步骤自报：1 次请求（1 次失败），0.000 美元，42181 个令牌输入、3875 个输出，97 秒。

## 关于证明器和模型的说明

分析者为目标命名的内容：

| 步骤 | 目标 | 特性 | 理由 |
|---|---|---|---|
| 21 | Z01, Z03, A03, A08, D01, K01, A03.7, A08.8, Z01.9, X01 | many_sessions, other | 这些性质针对任意数量的存储与检索会话提出，ADRF 记录存储被建模为单调表，恶意消费者被建模为持有自己凭据的进程。未建模任何密码原语：SBI 由每个已注册 NF 的一条私有通道表示（TS 29.500/33.501 的 TLS/OAuth2 保护），因此没有简化任何代数定律，答案也不依赖异或、Diffie-Hellman 或计数器。没有目标依赖一方持有并变化的值，因此不需要 bounded.pv。 |

## 分析者的结语

对 ADRF ML 模型存储建模了一次存储事务及其检索（TS 29.520 4.3.2.2.2、4.3.2.3.2、5.2.3.2.3.1/2、5.2.6.2.2/2.3、5.2.9），并回答了全部十个目标，每个主张都在目标所指名的攻击者之下给出。带对照的证明结果：201 应答消费者自己的存储请求，并以 nfInstanceId 携带请求方；200 仅发给存储的 NWDAF 或模型允许清单上的消费者，且仅发给有效令牌持有者，无令牌的客户端从不获得服务；所判定的身份就是发送请求的身份；检索到的条目与所请求的 modelUniqueId 匹配；访问令牌未被暴露。发现一个文稿未说明的弱点：任何持有单一服务范围令牌的已注册 NF 消费者，都可以用自己选择的 mlFileAddr 和一个列出目标消费者的允许清单，在现有 modelUniqueId 下注册第二条记录，而 ADRF 会保留两条记录并可能提供恶意的那条，于是获授权的消费者下载了攻击者选定的文件（s0048 会为已存储的模型保留新记录；文本中没有任何内容将标识符与其存储者绑定）。解读、限制和一张攻击图已记录；建议（每个 modelUniqueId 只允许一条关联、409 Conflict、原子检查）已给出，但未用 check_fix 检查，因为该失败取决于到达顺序，而合理的修改位于所建模的消息流之外。结果依赖于以下假设：3GPP 服务化接口经过认证并受完整性保护（TS 29.500/33.501，不在所选文本之内），以及将 s0083 的 403 条件解读为“在清单上或是存储的 NF”；未建模通过 store-trans-id 检索、多模型记录、部分失败、更新和删除。