NWDAF 分析订阅生命周期的安全分析

本报告的其他语言版本：English。

19.8.0 · ProVerif+ Agent 报告

文稿	5G System; Network Data Analytics Services; Stage 3（协议定义）
检查的性质	12 条：4 条成立，3 条不成立（3 个攻击），5 条无答案
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

本协议是 TS 29.520 中的 Nnwdaf_EventsSubscription API：NF 服务消费者向 NWDAF 请求网络性能分析，并在它提供的回调 URI 上接收通知。参与方包括消费者、生成分析的 NWDAF，以及作为 OAuth2 授权服务器、为该 API 签发访问令牌的 NRF。

一次运行对应一次订阅：消费者获取令牌，携带其 notificationURI 和事件提交订阅，NWDAF 创建资源并返回分配的 subscriptionId，向该 URI 发送至多两条通知，之后在消费者发送 DELETE 时删除该订阅。各方最终都持有该单个资源的 subscriptionId，并且只有该消费者能读取通知中的分析数据。

参与方：

- **NF 服务消费者**：分析数据的消费者；持有客户端凭据。持有：客户端凭据、要订阅的事件、notificationURI。
- **NWDAF**：生成分析数据；持有订阅存储。除自身身份外不持有其他内容。
- **NRF**：授权服务器。持有：已注册的 NF 身份。

图 1 展示了文稿规定的交互流程。

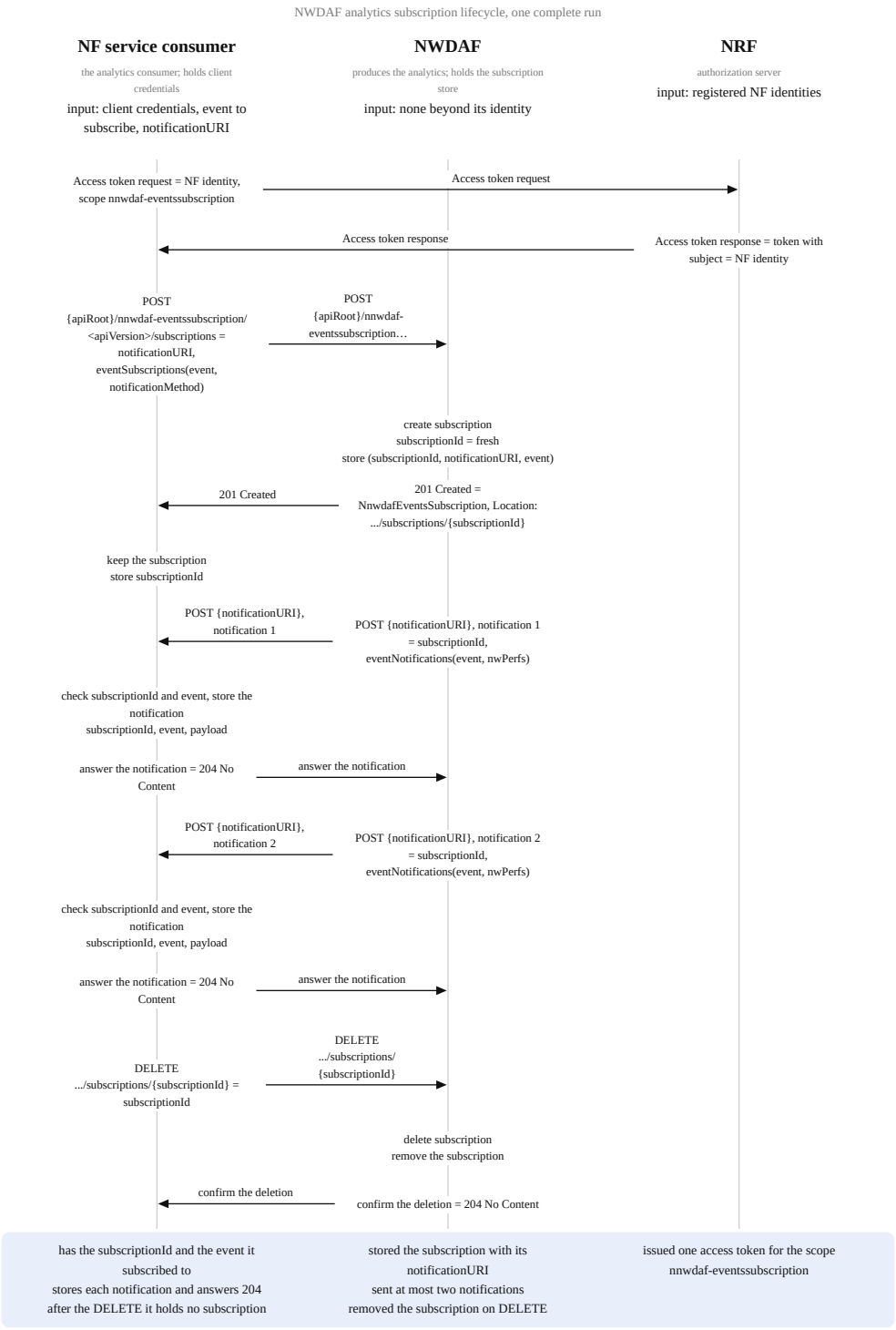


图 1. 文稿规定的协议。

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

分析共有 1 项发现：以下内容文稿中并未写明。其中 1 项无法确定文稿是否写明。

该交互针对网络成立：没有 NRF 签发的令牌，任何操作都不被接受；通知经过认证、绑定到发起订阅的消费者并且保密。分析发现了一处文稿未写明的弱点：NWDAF 从不检查更新或删除订阅的消费者是否就是创建该订阅的消费者，因此另一个持有同一 scope 有效令牌的 NF，只要持有该 subscriptionId，就能取消他人

的订阅。已检查的补救措施是将令牌主体与资源一同记录，并拒绝来自任何其他消费者的 Update 或 Unsubscribe。十二个目标中有五个都涉及变更或取消之后的生命周期，这些目标只能通过可达性来回答，因为删除与替换无法用 ProVerif 的单调状态表示；其余七个目标各有证明或攻击，并附带各自的对照查询。

攻击图沿用图 1 的列。红色表示攻击者的行为。黄色标记的是：某一方发送了消息，而规定流程在该位置并无此消息；以及攻击所破坏的结果。

2.1 另一个 NF 可以取消并非由它创建的订阅

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 (ts-33-501:2875)
价值	中
涉及的性质	Z03, Z03.10, A03（见第 3 节）
原文	s0278, s0280, s0324, s0325
依据	ProVerif attack queries `own_del`, `del_a`; Figures 2, 3 and 4

发生了什么。 NWDAF 接受任何持有该 API 有效令牌的 NF 对 Individual NWDAF Event Subscription 发起的 HTTP DELETE，并删除请求所命名的订阅。原文中没有任何内容要求调用方必须是创建该资源的消费者，模型中也同样没有这项检查。订阅者在自己从未请求取消的情况下就失去了订阅。

攻击者需要什么。 一个持有自身客户端凭据、并拥有 nnwdaf-eventssubscription scope 访问令牌的合法 NF，以及受害者的 subscriptionId。原文既没有把该标识符绑定到其创建者，也没有要求它不可预测。

攻击者能得到什么。 发往受害者的分析数据投递停止。利用同一缺口，Update 可以将 notificationURI 指向攻击者，这还会把受害者订阅的分析数据交给攻击者。

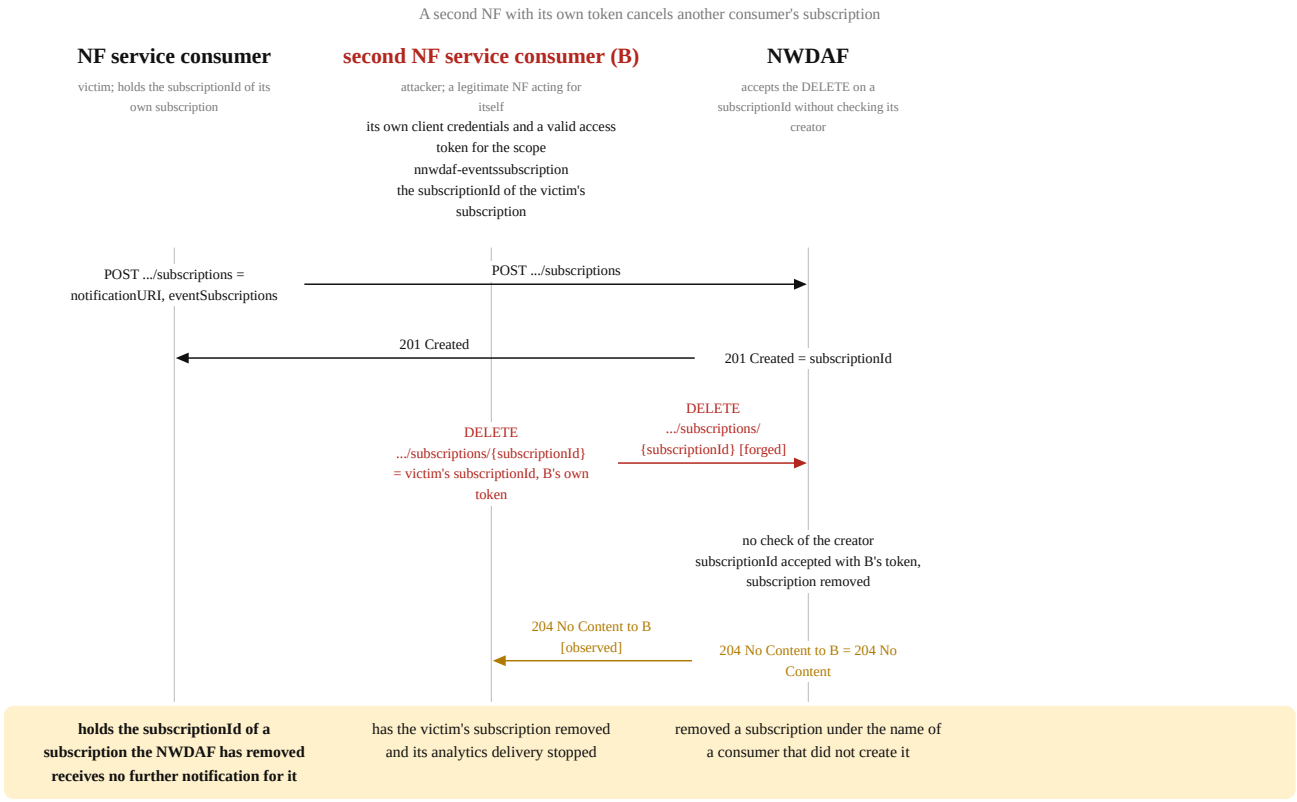


图 2. 另一个持有自身令牌的 NF 取消了另一消费者的订阅。性质 Z03；攻击查询 own_del。

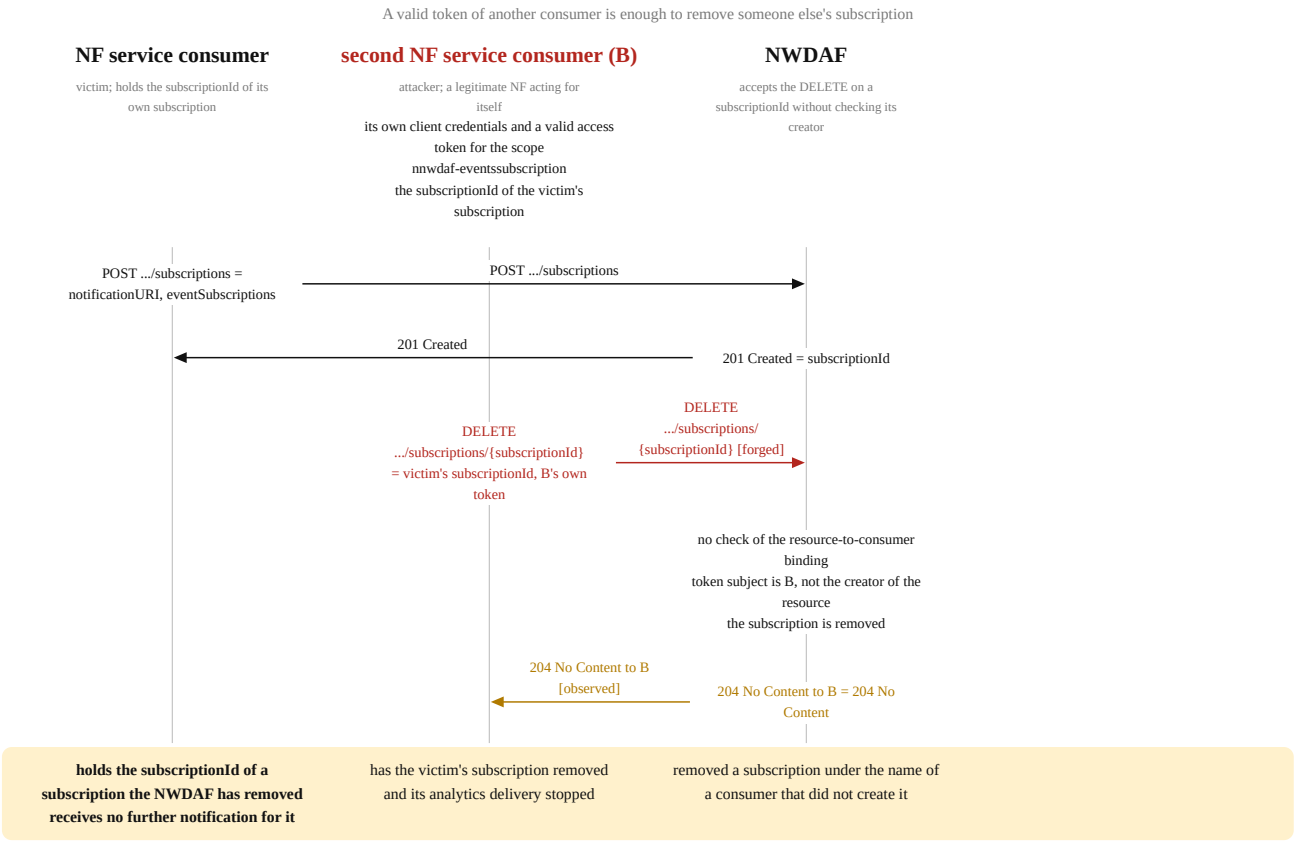


图 3. 另一消费者的有效令牌足以删除他人的订阅。性质 Z03.10；攻击查询 own_del。

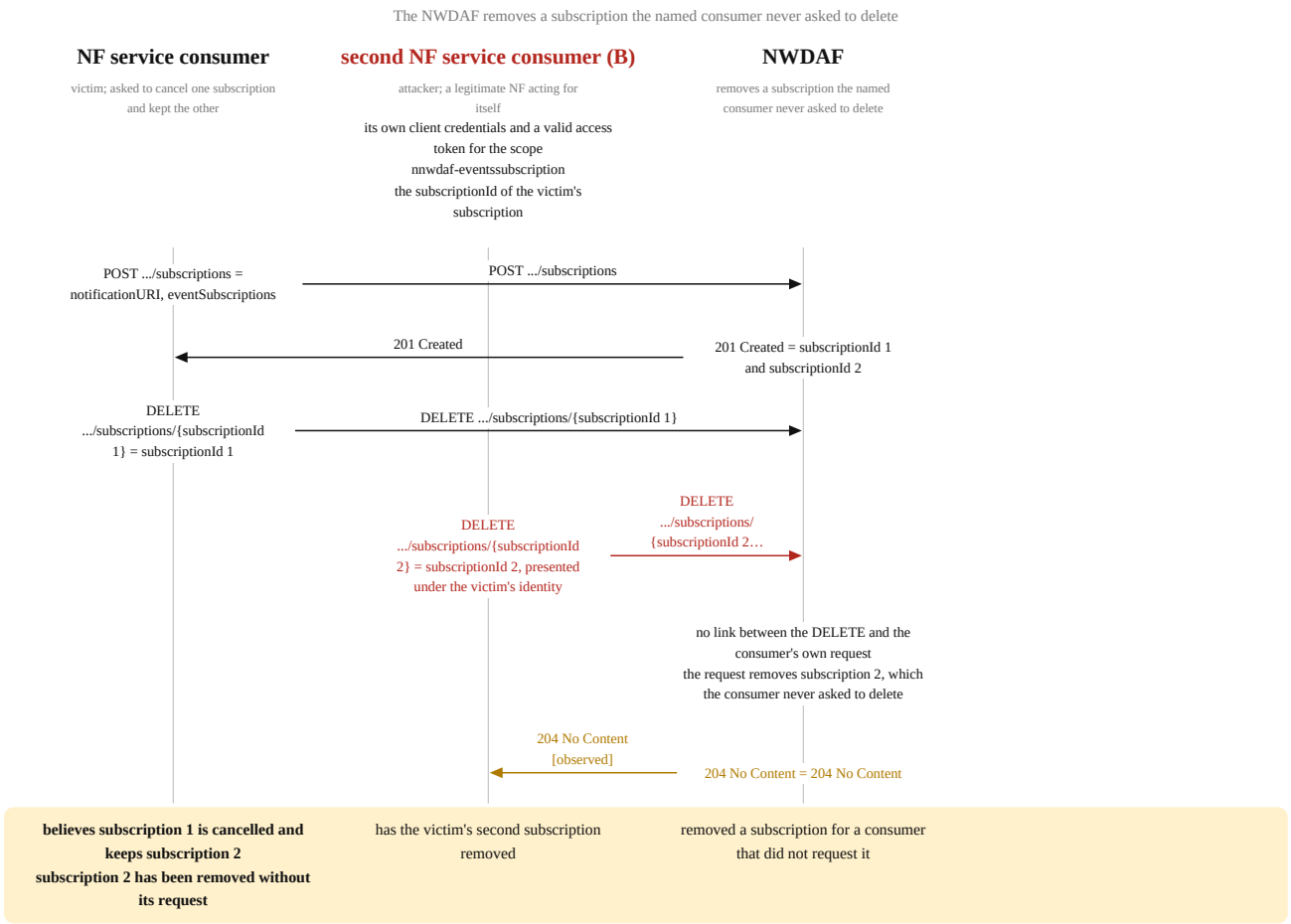


图 4. NWDAF 删除了一个被指名的消费者从未请求删除的订阅。性质 A03；攻击查询 del_a。

拟议修改。 在创建订阅时把 OAuth2 令牌的主体与订阅一同记录，并在 Update 和 Unsubscribe 时拒绝请求，除非令牌主体正是所记录的那个消费者。该修改已在模型的副本上检查：加上它之后，使另一个 NF 能够删除订阅的已记录攻击不再被发现，正常流程仍被发现，其他已记录的答案均无变化。

用 ProVerif 复核。 该修改已放入模型的副本：把创建订阅的消费者（OAuth2 令牌的主体）与资源一同记录，并在 Unsubscribe 时（以同样方式在 Update 时）拒绝请求，除非令牌主体正是所记录的那个消费者。在修改后的模型上，ProVerif 证明了 `own_del`：该攻击在那里不再可能。协议的正常流程仍被发现。其他 13 个已记录的答案不变。这是对模型的首次检查：它表明该修改在模型中消除了这一攻击，而非表明它不会引入其他问题。修改后的模型：`records/fixes/0055-fix-binding.pv`。

这些攻击图没有为某发现的这些攻击提供对应图：A03: own_del, Z03.10: del_a, Z03: del_a。

3 检查过的性质

每一行都是提交给 ProVerif 的一条性质。结果即 ProVerif 针对分析者所建协议模型给出的答案；价值和文稿是否已写明该结果属于分析者的判断，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	**每个操作的OAuth2授权。** NWDAF 每次提交受保护操作（创建、更新、取消订阅，或向消费者发送通知）都必须有授权服务器（NRF）签发的、对该 NF 消费者与 Nnwdaf_EventsSubscription API 有效的 OAuth2 令牌；当本地策略要求用户同意时，未获同意的目标 UE 不得被纳入分析收集。	无	成立	低	是

ID	性质	假定的泄露	结果	价值	文稿是否已写
Z03	**订阅绑定到其消费者。** OAuth2 令牌的授权主体、受众（NWDAF 实例）与 scope 必须与所执行的操作一致；订阅的 subscriptionId 与 notificationURI 必须绑定到创建该订阅的 NF 消费者，通知只能发往该订阅当前绑定的 notificationURI，不能因令牌或订阅移植而把权限或通知移到错误的消费者或操作上。	无	不成立（1 个攻击）	中	不确定
D01	**仅接受来自其NWDAF的通知。** NF 消费者接受的每条事件通知都对应其订阅的 NWDAF 真实发送的通知，subscriptionId、事件标识与通知内容一致，未被攻击者替换或篡改。	无	成立	低	是
A03	**消费者与NWDAF就订阅达成一致。** NF 消费者接受订阅创建/更新响应时，存在 NWDAF 的真实运行，双方对 subscriptionId、eventSubscriptions 与 notificationURI 一致；NWDAF 接受取消请求时，对应消费者确实请求删除该 subscriptionId。	无	不成立（1 个攻击）	低	不确定
A04	**每次创建使用不同的subscriptionId。** 每次成功的订阅创建对应不同的 NWDAF 运行与不同的 subscriptionId；同一 subscriptionId 的删除至多生效一次，重放的创建或删除请求不能产生额外的订阅实例或删除他人订阅。	无	只确认可达	低	否
R02	**每个请求和通知仅被接受一次。** 同一逻辑订阅请求（POST/PUT/DELETE）或同一逻辑通知在接收端至多被接受一次，重传的报文不会造成重复订阅、重复更新或重复存储通知。	无	只确认可达	低	否
S05	**取消后无通知。** 订阅被取消（DELETE 成功）或 NWDAF 以 termCause 终止之后，该 subscriptionId 不再产生新的通知，也不再接受更新；旧上下文不能重新获得原订阅的权限。	无	只确认可达	低	否
S04	**更新后的订阅用于后续通知。** 更新之后的通知使用更新后的 notificationURI 与订阅参数；静音/恢复状态在更新与通知之间一致继承；后续阶段不会使用已被替换的旧上下文。	无	只确认可达	低	否
K02	**分析负载仅发送给其订阅者。** 事件通知载荷（eventNotifications 中的分析数据）只对订阅该事件的 NF 消费者与其 NWDAF 可见，其他 NF（包括持有自身凭据的合法 NF）不能读取。	无	成立	低	是
Z03.10	**令牌主体与订阅所有权。** OAuth2 令牌的授权主体、受众与 scope 必须与所执行操作一致；subscriptionId 与 notificationURI 必须绑定到创建该订阅的 NF 消费者，通知只发往该订阅当前绑定的 notificationURI。	无	不成立（1 个攻击）	中	不确定
U01	**通知不跨订阅。** 消费者接受的每条通知的 subscriptionId 属于该消费者自己的订阅，通知中的事件数据由该订阅的 NWDAF 为该订阅生成，不与其他订阅或消费者交叉。	无	成立	低	是
U02	**更新或取消后无过期通知。** 订阅更新或取消生效之后，消费者不再接受按旧订阅参数生成的通知；仅更新或取消生效前已发出的在途通知可被接受。	无	只确认可达	低	否

所提出的问题及分析者的回答：

问题	回答	依据
Q1: 订阅 ID、notificationURI 与消费者身份绑定	所选原文中没有任何内容将 subscriptionId 或 notificationURI 绑定到创建它的消费者：来自另一个 NF 的同一 scope 令牌就足以取消该 NF 所持有的 subscriptionId，已检查的补救措施是把令牌主体与资源一同记录，并在 Update 和 Unsubscribe 时进行比较。	Z03, Z03.10, Z01
Q2: 通知相关性和跨订阅数据隔离	通知通过 subscriptionId 关联，该标识符受 NWDAF 签名覆盖并绑定到发起订阅的消费者，载荷只有该消费者可读，因此数据不会跨订阅；将 notificationURI 重新指向的更新未被建模。	U01, D01, K02
Q3: 更新和有限状态取消后陈旧通知的接受条件	对于更新或取消后的陈旧通知，只确认了可达性；要给出答案，需要一个能展开订阅状态的模型，而当前模型用单调表无法做到这一点。	U02, S05, S04

4 本次分析的局限

模型上的结果只说明：按规定的消息与检查，在把密码学视为理想的前提下，能给出什么、不能给出什么，与实现无关。

本次分析未覆盖的内容，按分析者的表述：

- TLS 被理想化为逐端点的签名与理想原语下的公钥加密；不声称计算安全性或抗量子性，也不建模令牌与 TLS 会话之间的通道绑定。
- 更新或取消之后的订阅生命周期未建模：ProVerif 的表只增不减，因此订阅实际上永远不会被移除，muting、replacement 与 termCause 状态也不存在。五个生命周期目标的答案仅由可达性给出。
- 更新操作（PUT）未单独建模；它对 notificationURI 的影响依据原文讨论。
- 通知由展开的进程计数（每个订阅两条），而不是由 maxReportNbr 检查计数，因此不分析报告数量上限。
- 用户同意、UDM 交互、订阅转移、漫游、SCP 或间接通信以及重定向都在所选流程之外。
- 攻击者从不持有已存储的订阅状态：恶意 NF 的建模方式是给攻击者一个 NF 的凭据，而不是让它读取订阅存储。

宿主记录的内容：

- 无答案：A04、R02、S05、S04、U02。
- 在解读写成之后答案发生变化的：Z01、U01。
- 分析者为性质设定了攻击者条件，但既没有证明也没有攻击的（附录 D）：Z03 在 `network` 下; A04 在 `network` 下; A04 在 `malicious_nf` 下; R02 在 `network` 下; R02 在 `malicious_nf` 下; S05 在 `network` 下; S05 在 `malicious_nf` 下; S04 在 `network` 下; S04 在 `malicious_nf` 下; U02 在 `network` 下; U02 在 `malicious_nf` 下。
- 没带对照查询的证明（附录 C）：Z01: auth_sub, K02: payload_secret。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。

每条性质还有哪些问题悬而未决，以及什么能解决它们，见附录 A。

附录 A：逐条性质的解读

这是分析者自己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，能否按现状使用，以及它有多大价值。

Z01: 每个操作的OAuth2授权

NWDAF 每次提交受保护操作（创建、更新、取消订阅，或向消费者发送通知）都必须有授权服务器（NRF）签发的、对该 NF 消费者与 Nnwdaf_EventsSubscription API 有效的 OAuth2 令牌；当本地策略要求用户同意时，未获同意的目标 UE 不得被纳入分析收集。

本目标的答案在写成此解读之后发生了变化。

NWDAF 接受的每次 Subscribe 和 Unsubscribe 之前，都有一枚 NRF 签发给同一 NF、scope 为 nnwdaf-eventssubscription 的访问令牌，没有令牌的请求会失败。恶意 NF 可以调用该 API，但只能以自己的名义。目标的用户同意那一半并不出现：此流程是非 UE 的聚合网络性能，不携带 SUPI 或 GPSI，因此 NWDAF 从不进入同意检查。

- 结果：成立；回答了所问的问题：是；可带保留使用
- 价值低：该要求正是 5.1.9 和 TS 33.501 所规定的；分析机械地复现了它，未找到绕过途径。
- 文稿中有规定：是。文稿中的位置：ts-33-501:2885, s0565, s0568, s0570
- 依据：令牌被建模为由 NRF 签名、由 NWDAF 校验的三元组；未建模过期、受众与吊销。；NRF 在签发令牌之前对客户端进行认证，原文将其归于 TS 29.510。
- 待解决：令牌过期、受众与吊销；为某个 API 签发的令牌能否在另一个 API 上重放
- 下一步：建模令牌生命周期与受众，并询问相同的性质。

Z03: 订阅绑定到其消费者

OAuth2 令牌的授权主体、受众（NWDAF 实例）与 scope 必须与所执行的操作一致；订阅的 subscriptionId 与 notificationURI 必须绑定到创建该订阅的 NF 消费者，通知只能发往该订阅当前绑定的 notificationURI，不能因令牌或订阅移植而把权限或通知移到错误的消费者或操作上。

通知是有绑定的：只有订阅所属的消费者能读取它们，每条通知都携带 subscriptionId 和该消费者所绑定的事件。归属那一半不成立：只要任何 NF 出示有效令牌和正确的 subscriptionId，NWDAF 就会移除该订阅，因为原文没有规定把资源与其创建者绑定起来的规则。

- 结果：不成立（1 个攻击）；回答了所问的问题：部分；可带保留使用
- 价值中：TS 33.501 要求每个 NF 校验收到的消息，而 API 文本只说 NWDAF 移除请求中所指的订阅；两者都没有陈述这一弱点，也没有陈述能防止它的绑定。
- 文稿中有规定：不明确。所依据的原文：s0280, s0324, s0325, s0327
- 依据：该攻击需要 subscriptionId；原文只要求 NWDAF 分配一个，并未要求它不可猜测或仅为所有者所知。；未建模更新，因此同一原因导致的 notificationURI 重定向是依据原文论证的，而非由证明器展示。
- 待解决：subscriptionId 在何处为另一个 NF 所知；陌生者的更新是否被拒绝
- 下一步：建模更新，并建模 SCP 或另一个 NF 可观察到的标识符，以确定标识符泄露的条件。

D01: 仅接受来自其NWDAF的通知

NF 消费者接受的每条事件通知都对应其订阅的 NWDAF 真实发送的通知，subscriptionId、事件标识与通知内容一致，未被攻击者替换或篡改。

消费者只接受 NWDAF 为该消费者、为该消费者持有的订阅、并针对其所订阅事件签名的通知。网络攻击者既不能伪造也不能篡改通知，为另一个消费者构造的通知会被拒绝。

- 结果：成立；回答了所问的问题：是；可带保留使用
- 价值低：TS 33.501 要求 SBI 上的双向认证和对收到消息的校验，这正是该证明所复现的。
- 文稿中有规定：是。文稿中的位置：ts-33-501:2871, s0340, s0383, s0388
- 依据：理想签名和理想公钥加密代替双向 TLS；未建模 NWDAF 身份与其被访问名称之间的证书绑定。；消费者仅按 subscriptionId 和事件进行关联；可选的 notifCorrId 未建模。
- 待解决：在活跃订阅内重放通知；消费者在取消后是否接受通知
- 下一步：在展开消费者订阅状态的模型上询问重放问题。

A03: 消费者与NWDAF就订阅达成一致

NF 消费者接受订阅创建/更新响应时，存在 NWDAF 的真实运行，双方对 subscriptionId、eventSubscriptions 与 notificationURI 一致；NWDAF 接受取消请求时，对应消费者确实请求删除该 subscriptionId。

消费者的 201 Created 是真实的：它存储的 subscriptionId 是 NWDAF 在真实运行中创建的，由网络攻击者构造的响应会被拒绝。取消那一半并不以同样方式成立：NWDAF 可能接受归属于某消费者的取消，而该消费者从未请求删除该 subscriptionId，因此双方并不总是就所移除的内容达成一致。

- 结果：不成立（1 个攻击）；回答了所问的问题：部分；可带保留使用
- 价值低：创建上的一致遵循原文；取消上的不一致与 Z03 发现所描述的同一缺口相同，原文并未陈述它。
- 文稿中有规定：不明确。所依据的原文：s0278, s0282, s0324, s0327
- 依据：响应通过签名载荷内的身份与请求的消费者绑定，该身份代替 TLS 会话。；订阅状态在模型中不会被移除，因此不分析多次变更之后的一致。
- 待解决：更新之后的一致；DELETE 重传时的一致
- 下一步：展开创建、更新与取消，并将订阅状态保存在进程链中。

A04: 每次创建使用不同的subscriptionId

每次成功的订阅创建对应不同的 NWDAF 运行与不同的 subscriptionId；同一 subscriptionId 的删除至多生效一次，重放的创建或删除请求不能产生额外的订阅实例或删除他人订阅。

存在带有自己 subscriptionId 的创建运行，这就是记录答案所显示的全部内容。重放的创建是否增加第二个实例、重复的删除能否移除第二个订阅，均未判定：模型无法对实例计数，也无法移除订阅。

- 结果：仅为可达；回答了所问的问题：部分；可带保留使用
- 价值低：原文未规定幂等规则，因此没有可复现的规定；搜索也未发现对弱点的陈述。
- 文稿中有规定：否。所依据的原文：s0280, s0325
- 依据：只运行了可达性；关于实例数量或重放的任何性质都没有记录答案。；subscriptionId 是新鲜名称，因此标识符的互异性是模型内置的，而非经检验的。
- 待解决：重传的 POST 是否创建第二个订阅
- 下一步：展开两次创建和两次删除，并询问一个单射性质。

R02: 每个请求和通知仅被接受一次

同一逻辑订阅请求（POST/PUT/DELETE）或同一逻辑通知在接收端至多被接受一次，重传的报文不会造成重复订阅、重复更新或重复存储通知。

生成、投递并接受了一条通知，因此目标所谈论的正常流程存在。某个请求或通知是否会被接受两次尚未判定，因为重复需要一种 ProVerif 无法在无界数量的会话之间保持的状态。

- 结果：仅可达；是否回答了所问：部分；可在附带注意事项的情况下使用
- 价值低：原文没有给出事务标识符，也没有提到重复；没有找到可复现的陈述。
- 文稿中是否陈述：否。所依据的原文：s0282、s0388
- 依据：只运行了可达性。；HTTP 重传发生在所建模的 SBI 之下，因此模型中没有可抑制的重复。
- 未决：重复的通知对消费者存储的影响
- 下一步：将消费者展开到两个接受步骤，并询问同一条通知是否会被存储两次。

S05: 取消后无通知

订阅被取消（DELETE 成功）或 NWDAF 以 termCause 终止之后，该 subscriptionId 不再产生新的通知，也不再接受更新；旧上下文不能重新获得原订阅的权限。

为某个订阅发送了一条通知，该订阅随后被取消，因此该情形可达。取消生效后是否仍能产生或接受通知尚未判定，因为模型无法删除订阅，因而无法表示删除的瞬间。

- 结果：仅可达；是否回答了所问：部分；可在附带注意事项的情况下使用
- 价值低：原文陈述了删除（s0325..s0327）和 termCause 指示（s0386），但没有提到在途通知。
- 文稿中是否陈述：否。所依据的原文：s0325、s0327、s0386
- 依据：只运行了可达性。；NWDAF 在已记录取消时会抑制通知，但模型的记录只增不减，因此这接近于定义而非经过检查的性质。
- 未决：在 DELETE 之前产生、在其之后投递的通知；以 termCause 终止而非 DELETE
- 下一步：将 NWDAF 展开为记录取消的链，并询问第三条通知是否可以在其后出现。

S04: 更新后的订阅用于后续通知

更新之后的通知使用更新后的 notificationURI 与订阅参数；静音/恢复状态在更新与通知之间一致继承；后续阶段不会使用已被替换的旧上下文。

创建了一个订阅并接受了为其发送的通知，因此正常流程存在。后续通知是否遵循更新后的参数，以及被替换的 notificationURI 是否停止接收，尚未判定，因为更新和替换无法在模型的状态中表示。

- 结果：仅可达；是否回答了所问：部分；可在附带注意事项的情况下使用
- 价值低：原文陈述了订阅会被更新并存储、notificationURI 可以被修改；但没有提到在途通知，因此没有可复现的陈述结果。
- 文稿中是否陈述：否。所依据的原文：s0300、s0303、s0306
- 依据：只运行了可达性。；更新操作未建模；解读仅限于 s0300..s0303 所述内容。
- 未决：更新后发往旧 notificationURI 的通知是否仍会被接受
- 下一步：将创建、更新和两条通知展开，并把 URI 作为链的状态。

K02: 分析负载仅发送给其订阅者

事件通知载荷（eventNotifications 中的分析数据）只对订阅该事件的 NF 消费者与其 NWDAF 可见，其他 NF（包括持有自身凭据的合法 NF）不能读取。

通知中的分析数据只能由该订阅所属的消费者读取。网络攻击者和持有自身凭据的另一个合法 NF 都无法获知载荷，且对照查询表明攻击者所控制的 NF 的载荷是已知的，因此结果并非空洞成立。

- 结果：Holds；是否回答了所问：是；可在附带注意事项的情况下使用
- 价值低：TS 33.501 要求 SBI 信令的机密性，这正是证明所复现的内容。
- 文稿中是否陈述：是。文稿中的出处：ts-33-501:2871、s0340、s0565
- 依据：用理想公钥加密代替 TLS；通知被加密至订阅的消费者，这是模型对端点认证 TLS 会话的替代。；载荷是单个不透明值；其内容未建模。
- 未决：当 notificationURI 指向另一个 NF 的端点时的机密性
- 下一步：将回调 URI 建模为 NWDAF 解析的地址，并询问另一个 NF 能否使订阅投递到该地址。

Z03.10: 令牌主体与订阅所有权

OAuth2 令牌的授权主体、受众与 scope 必须与所执行操作一致；subscriptionId 与 notificationURI 必须绑定到创建该订阅的 NF 消费者，通知只发往该订阅当前绑定的 notificationURI。

令牌部分成立：发给一个 NF 的令牌不能被另一个 NF 用来使操作被接受，且令牌的主体与请求身份进行比较。所有权部分与 Z03 中完全一样不成立：订阅资源没有与创建它的消费者绑定，因此具有相同 scope 的有效令牌就足以对另一个消费者的订阅进行操作。

- 结果：Fails（1 个攻击）；是否回答了所问：部分；可在附带注意事项的情况下使用
- 价值中：RFC 6749 指出 OAuth2 未给资源服务器提供检查令牌是否签发给出示它的客户端的方法，TS 33.501 要求每个 NF 验证传入消息；两者均未陈述此 API 的这一弱点，也未陈述资源到消费者的绑定缺失。
- 文稿中是否陈述：不明确。所依据的原文：rfc6749:2879、s0568、s0570、s0324
- 依据：令牌主体与请求中的普通身份比较，而非与 TLS 客户端证书比较；原文将该绑定留给 TS 29.500。；攻击需要 subscriptionId，而原文并未要求其保密。
- 未决：受众或按资源的 scope 能否弥补该缺口；陌生者的更新是否会被拒绝
- 下一步：添加指明该资源的受众值，并测试仅凭它能否阻止另一个消费者的请求。

U01: 通知不跨订阅

本目标的答案在本解读写出后发生了变化。

通知仅为 NWDAF 为同一消费者创建的 subscriptionId 产生，且消费者仅为其自身拥有的、其所订阅事件的订阅接受该通知。为一个消费者产生的通知不能在另一个消费者的 subscriptionId 下投递。

- 结果：Holds；是否回答了所问：是；可在附带注意事项的情况下使用
- 价值低：这是将 TS 33.501 的完整性与认证要求应用于通知，加上原文强制要求的 subscriptionId 关联。
- 文稿中是否陈述：是。文稿中的出处：ts-33-501:2875、s0383、s0388
- 依据：用理想签名和加密代替按端点的 TLS 会话。；消费者对自身订阅的存储是单调表，因此永远不会忘记已取消的订阅。
- 未决：消费者取消两个订阅之一后的跨订阅投递
- 下一步：展开同一消费者的两个订阅，并再次提出隔离问题。

U02: 更新或取消后无过期通知

订阅更新或取消生效之后，消费者不再接受按旧订阅参数生成的通知；仅更新或取消生效前已发出的在途通知可被接受。

为活跃订阅接受了一条通知，因此正常流程存在。在更新或取消生效后，按旧参数生成的通知是否仍会被接受尚未判定：模型无法删除或替换订阅，因此参数改变的瞬间未被表示。

- 结果：仅可达；是否回答了所问：部分；可在附带注意事项的情况下使用
- 价值低：原文给出了更新和删除何时生效，但没有针对在途通知的规则，因此没有可复现的陈述。
- 文稿中是否陈述：否。所依据的原文：s0300、s0325、s0386
- 依据：只运行了可达性。；消费者依据只增不减的表进行接受，因此永远不会因为订阅结束而拒绝通知。
- 未决：跨越取消瞬间的在途通知；跨越 notificationURI 更新的在途通知
- 下一步：将消费者和 NWDAF 展开为三个步骤，并在链中携带订阅状态。

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，以及各攻击及分析者对其的判定。

目标	级别 界	查询	对照	攻击	主张
Z01	proved	auth_sub	6 项中的 4 项		6 项 proved
Z03	proved	payload_secret	checked	own_del (depends_on_ambiguity)	1 项 proved，1 项 reachable，1 项 attack
D01	proved	auth_notif	checked		2 项 proved
A03	proved	sub_a	checked	del_a (depends_on_ambiguity)	1 项 proved，1 项 attack
A04	reachable	ctl_bcreate			3 项 reachable
R02	reachable	reach_notif			3 项 reachable
S05	reachable	ctl_payload			3 项 reachable
S04	reachable	reach_notif			3 项 reachable
K02	proved	payload_secret	3 项中的 2 项		3 项 proved
Z03.10	proved	auth_del	checked	own_del (depends_on_ambiguity)	2 项 proved，1 项 attack
U01	proved	auth_notif	checked	notif_own (model_artifact)	3 项 proved，1 项 attack
U02	reachable	reach_notif			3 项 reachable

逐条主张，按分析者对每条主张的表述；每条的回答来自证明器：

目标	主张	攻击者	回答	查询	对照
Z01	NWDAF 接受的每个 Subscribe 之前，NRF 都已针对 nnwdaf-eventssubscription 范围向同一个 NF 签发过 OAuth2 访问令牌；网络中的攻击者不持有任何一方的秘密。	`network`	proved	auth_sub	checked
Z01	NWDAF 接受的每个 Subscribe 之前，都已向同一个 NF 签发过 OAuth2 访问令牌，请求者是持有自己有效令牌的恶意 NF 时也是如此；这样的 NF 可以订阅，但只能以自己的名义订阅。	`malicious_nf`	proved	auth_sub	none
Z01	用户同意主张：所选流程是非 UE 的聚合网络性能，不携带 SUPI 或 GPSI，因此 4.2.2.2.2 的同意关卡永远不会进入；取而代之的是，没有 NRF 向该 NF 签发的令牌，任何 Subscribe 都不会被接受。	`network`	proved	auth_sub	none
Z01	NWDAF 接受的每个 Subscribe 之前，都已向同一个 NF 签发过令牌，请求者是持有自己有效令牌的恶意 NF 时也是如此。	`malicious_nf`	proved	auth_sub	checked
Z01	用户同意主张：所选流程不携带 SUPI 或 GPSI，因此不会进入同意关卡；取而代之的是，没有向该 NF 签发的令牌，任何 Subscribe 都不会被接受。	`network`	proved	auth_sub	checked

目标	主张	攻击者	回答	查询	对照
Z01	NWDAF 接受的每个 Subscribe 之前，NRF 都已针对 nnwdaf-eventssubscription 范围向同一个 NF 签发过 OAuth2 访问令牌。关于攻击者的说明：模型公开了第二个 NF 的客户端密钥，因此本文件中的攻击者可以充当一个合法 NF；所以这个回答针对的是拥有网络外加一个 NF 凭据的攻击者，这是两种情形中较强的一种。	network	proved	auth_sub	checked
Z03	某个订阅的通知只能由该订阅所属的消费者读取，并且携带该消费者绑定的 subscriptionId 和事件：另一个持有自己凭据的 NF 无法读取或接收该通知。	malicious_nf	proved	payload_secret	checked
Z03	只有创建了某个订阅的 NF 服务消费者才能用自己有效的令牌删除该订阅。	malicious_nf	attack（depends_on_ambiguity）	own_del	
Z03	某个订阅及其通知可达，因此重要的订阅/通知绑定问题是针对一次真实运行提出的。	network	reachable	reach_notif	
D01	NWDAF 产生的每条通知都是针对它为同一个消费者创建的订阅产生的，由 NWDAF 签名并绑定到该消费者，且携带该消费者所订阅的事件；网络攻击者既无法伪造也无法篡改它。	network	proved	auth_notif	checked
D01	每条通知都是针对为同一个消费者创建的订阅产生的，请求者是恶意 NF 时也是如此。	malicious_nf	proved	auth_notif	checked
A03	消费者在 201 Created 响应中接受的 subscriptionId 确实是 NWDAF 在该次运行中创建并分配的；由网络攻击者构造的响应会被拒绝。	network	proved	sub_a	checked
A03	当 NWDAF 接受某个消费者的取消请求时，该消费者确实请求删除了那个 subscriptionId。	malicious_nf	attack（depends_on_ambiguity）	del_a	
A04	存在一次订阅创建运行，其中带有 NWDAF 为该次运行分配的 subscriptionId；不同的运行是不同的会话。		reachable	ctl_bcreate	
A04	带有自己 subscriptionId 的一次订阅创建可达。	network	reachable	ctl_bcreate	
A04	带有自己 subscriptionId 的一次订阅创建可达，包括恶意 NF 以自己的名义发起的一次。	malicious_nf	reachable	ctl_bcreate	
R02	一条通知被产生、投递并被接受，因此本目标的重传问题是针对一次已达成的运行提出的。		reachable	reach_notif	
R02	一条通知被产生、投递并被接受。	network	reachable	reach_notif	
R02	针对攻击者不控制的消费者，一条通知被产生、投递并被接受。	malicious_nf	reachable	reach_notif	
S05	针对一个随后被取消的订阅发送了一条通知，因此取消后的问题是针对一次已达成的运行提出的。		reachable	ctl_payload	
S05	针对一个随后被取消的订阅发送了一条通知。	network	reachable	ctl_payload	
S05	针对攻击者不控制的一个消费者，为其随后被取消的订阅发送了一条通知。	malicious_nf	reachable	ctl_payload	
S04	创建了一个订阅，且针对它的通知被接受，因此更新后再通知的问题是针对一次已达成的运行提出的。		reachable	reach_notif	

目标	主张	攻击者	回答	查询	对照
S04	创建了一个订阅，且针对它的通知被接受。	network	reachable	reach_notif	
S04	针对攻击者不控制的一个消费者，创建了一个订阅，且针对它的通知被接受。	malicious_nf	reachable	reach_notif	
K02	发送给消费者的通知中的分析载荷不能被网络攻击者获知；对照表明，当载荷属于攻击者控制的 NF 时，攻击者确实知道该载荷。	network	proved	payload_secret	checked
K02	另一个合法 NF 即使持有自己的凭据并能调用 API，也无法读取发给另一个消费者的通知中的分析载荷。	malicious_nf	proved	payload_secret	none
K02	另一个持有自己凭据的合法 NF 无法读取发给另一个消费者的通知中的分析载荷。	malicious_nf	proved	payload_secret	checked
Z03.10	NWDAF 接受的 Unsubscribe 之前，都已向出示该请求的同一个 NF 签发过令牌：签发给一个 NF 的令牌不能被另一个 NF 用来在 API 上操作。	malicious_nf	proved	auth_del	checked
Z03.10	消费者操作的 subscriptionId 和 notificationURI 属于该消费者；任何令牌都不能把一项权利转移到另一个消费者的订阅上。	malicious_nf	attack（depends_on_ambiguity）	own_del	
Z03.10	NWDAF 接受的 Unsubscribe 之前，都已向同一个 NF 签发过令牌。	network	proved	auth_del	checked
U01	只有当 subscriptionId 是 NWDAF 为同一个消费者创建的时候，才会产生通知，因此另一个 NF 的通知无法以该消费者的 subscriptionId 到达该消费者。	malicious_nf	proved	auth_notif	checked
U01	每条通知都是针对为同一个消费者创建的订阅产生的；网络攻击者无法伪造一条通知。	network	proved	auth_notif	checked
U01	只有当 subscriptionId 是 NWDAF 为同一个消费者创建的时候，才会产生通知。	malicious_nf	proved	auth_notif	checked
U01	消费者只对它所认证的、带有其 201 响应的订阅接受通知；记录到的反例来自模型将双方共用一个存储，并不描述该协议。	malicious_nf	attack（model_artifact）	notif_own	
U02	针对一个存活的订阅，通知被接受，因此过期通知的问题是针对一次已达成的运行提出的。		reachable	reach_notif	
U02	针对消费者一个存活的订阅，通知被接受。	network	reachable	reach_notif	
U02	针对攻击者不控制的一个消费者的存活订阅，通知被接受。	malicious_nf	reachable	reach_notif	

附录 D：分析是怎么跑的

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

ProVerif+ Agent 会话 session-20261114T234020，2026-11-14。状态：已完成。模型：deepseek-v4-1-flash-261017。步骤：69，证明器 4 s，实际耗时 16 min，费用 0.129 USD。

ProVerif：运行 78 个证明，78 个已判定，0 个无答案，2 s，记录 38 个答案。

分析了什么

协议	NWDAF analytics subscription lifecycle
版本	19.8.0
分析范围	一个 NF 与 NWDAF、固定非 UE 聚合 network performance analytics；创建/更新/通知/取消单订阅，最多两次通知，排除 transfer/漫游/多 NWDAF。
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。 固定 Rel-19。分别区分外部网络攻击者与持有自身凭据的恶意合法 NF；TLS 认证加密按端点建模，合法 NF 不能读取其他 NF 的 TLS 会话。 OAuth2 subject/audience/scope/expiry 及 TLS 身份绑定依照所附基线；服务 scope 不自动等于资源所有权。单调表不能模拟真实删除，有限状态生命周期必须明确界定。目标是指定 API 子流程，不是完整 5G 安全。 本次仅修正文稿选择粒度：原 API 全章包含所有 analytics 类型导致超上下文，仍保留原协议范围和完整原稿可读性；未选的数值 payload 保持 opaque，不验证数值正确性。
原文	609 个带编号的原文单元（498330 字节），来自下列章节。本报告中的 s0001 这类引用即为这些原文单元之一

分析所依据的文稿章节：

文稿	章节	标题
main	4.2.2.2	Nnwdaf_EventsSubscription_Subscribe service operation
main	4.2.2.2.1	General
main	4.2.2.2.2	Subscription for event notifications
main	4.2.2.2.3	Update subscription for event notifications
main	4.2.2.3	Nnwdaf_EventsSubscription_Unsubscribe service operation
main	4.2.2.3.1	General
main	4.2.2.3.2	Unsubscribe from event notifications
main	4.2.2.4	Nnwdaf_EventsSubscription_Notify service operation
main	4.2.2.4.1	General
main	4.2.2.4.2	Notification about subscribed event
main	5.1.1	Introduction
main	5.1.2	Usage of HTTP
main	5.1.2.1	General
main	5.1.2.2	HTTP standard headers
main	5.1.2.2.1	General

文稿	章节	标题
main	5.1.2.2.2	Content type
main	5.1.2.3	HTTP custom headers
main	5.1.3.1	Resource Structure
main	5.1.3.2	Resource: NWDAF Events Subscriptions
main	5.1.3.2.1	Description
main	5.1.3.2.2	Resource definition
main	5.1.3.2.3	Resource Standard Methods
main	5.1.3.2.3.1	POST
main	5.1.3.2.4	Resource Custom Operations
main	5.1.3.3	Resource: Individual NWDAF Event Subscription
main	5.1.3.3.1	Description
main	5.1.3.3.2	Resource definition
main	5.1.3.3.3	Resource Standard Methods
main	5.1.3.3.3.1	DELETE
main	5.1.3.3.3.2	PUT
main	5.1.3.3.4	Resource Custom Operations
main	5.1.5	Notifications
main	5.1.5.1	General
main	5.1.5.2	Event Notification
main	5.1.5.2.1	Description
main	5.1.5.2.2	Operation Definition
main	5.1.6.2.2	Type NnwdafEventsSubscription
main	5.1.6.2.3	Type EventSubscription
main	5.1.6.2.4	Type NnwdafEventsSubscriptionNotification
main	5.1.6.2.5	Type EventNotification
main	5.1.6.2.7	Type EventReportingRequirement
main	5.1.6.2.22	Type NetworkPerfRequirement
main	5.1.6.2.23	Type NetworkPerfInfo
main	5.1.6.2.30	Type ThresholdLevel
main	5.1.6.2.49	Type ConsumerNfInformation
main	5.1.6.3.3	Enumeration: NotificationMethod
main	5.1.6.3.4	Enumeration: NwdafEvent
main	5.1.6.3.10	Enumeration: NetworkPerfType
main	5.1.6.3.26	Enumeration: TermCause
main	5.1.7	Error handling
main	5.1.7.1	General

文稿	章节	标题
main	5.1.7.2	Protocol Errors
main	5.1.7.3	Application Errors
main	5.1.8	Feature negotiation
main	5.1.9	Security

源文步骤在文稿中定位到的协议目的：一个 NF 与 NWDAF、固定非 UE 聚合 network performance analytics；创建/更新/通知/取消单订阅，最多两次通知，排除 transfer/漫游/多 NWDAF。

对照文稿定性质

在编写查询之前，分析者将起草的性质清单与文稿本身对协议目的的表述做了对照。名称、性质所涉及的内容以及文稿的表述均出自分析者的解读；所列位置确实存在于文稿中。

ID	简称	对象	文稿怎么说	出处
Z01	每个操作的OAuth2授权	NWDAF 确信：它从某个 NF 服务消费者接受的每一次 subscribe、update、unsubscribe 和 notification，都已由 NRF（授权服务器）代表该消费者针对 Nnwdaf_EventsSubscription API 授权；并且在本地策略要求时，目标 UE 已同意该分析采集。	5.1.9 说访问可由 OAuth2 Client Credentials 授权，NRF 作为授权服务器，并定义范围 nnwdaf-eventssubscription；4.2.2.2.2（s0287）要求 NWDAF 在本地策略要求时检查目标 UE 的用户同意，并在未获同意时以 USER_CONSENT_NOT_GRANTED 拒绝。	ts-33-501:2883, ts-33-501:2885, ts-33-501:2889, s0565, s0566, s0568, s0570, s0287
Z03	订阅绑定到其消费者	NWDAF 确信，且创建订阅的 NF 服务消费者也确信：只有该消费者才能更新或删除该订阅资源，通知只发往该订阅当前绑定的 notificationURI。	文稿定义了订阅资源、其 {subscriptionId} 和 notificationURI（s0303 NOTE 1 允许修改 notificationURI，以便把后续通知发给新消费者，s0324/s0325 按 subscriptionId 删除），并要求带范围的 OAuth2 授权，但没有规定调用方必须是创建该订阅的消费者。	ts-33-501:2875, rfc6749:2879, s0303, s0324, s0325, s0568, s0570
D01	仅接受来自其NWDAF的通知	NF 服务消费者确信：它接受的每一条事件通知都由其订阅的 NWDAF 发送，携带它订阅过的事件标识符，并且在途中未被修改。	4.2.2.4.2 规定 NWDAF 将通知 POST 到订阅时收到的 notificationURI（s0340），消费者存储该通知并应答 204（s0388..s0390）；5.1.9/s0565 和 TS 33.501 要求 SBI 请求/响应双向认证，且 SBI 信令受完整性保护。	ts-33-501:2871, ts-33-501:2875, s0340, s0383, s0388, s0390, s0565
A03	消费者与NWDAF就订阅达成一致	在创建、更新和取消之后，NF 服务消费者与 NWDAF 持有同一订阅：消费者接受的 subscriptionId 就是 NWDAF 创建的那个，NWDAF 接受的取消是由该消费者请求的。	4.2.2.2.2 s0278..s0282（创建、分配 subscriptionId、以表示和 Location 应答 201），4.2.2.2.3 s0300..s0306（更新），4.2.2.3.2 s0324..s0327（删除、应答 204）。	s0278, s0280, s0282, s0300, s0306, s0324, s0327

ID	简称	对象	文稿怎么说	出处
A04	每次创建使用不同的 subscriptionId	每次成功的订阅创建都获得自己的 subscriptionId 和资源，重放的创建或删除请求不能新增订阅实例，也不能删除另一个消费者的订阅。	s0280 “assign an event subscriptionId”，s0281 存储订阅，s0325..s0327 在 DELETE 时移除对应订阅；文稿自身没有幂等或重放规则。	ts-33-501:2875, s0278, s0280, s0281, s0325, s0327
R02	每个请求和通知仅被接受一次	NWDAF 和 NF 服务消费者对给定的创建、更新或删除请求，以及给定的通知，最多接受一次，因此重传不会产生重复订阅、重复更新或重复存储的通知。	服务操作是普通 HTTP POST/PUT/DELETE，响应为 201/200/204（s0282、s0304..s0306、s0325..s0327），通知是 POST 并以 204 应答（s0388..s0390）；文稿没有给出事务标识符或去重规则。	s0282, s0306, s0327, s0388, s0390
S05	取消后无通知	NWDAF 在 DELETE 时移除订阅，或以 termCause 终止订阅之后，该 subscriptionId 不再产生通知，也不再接受更新。	4.2.2.3.2 s0325..s0327（移除订阅），s0386（termCause：NWDAF 表示不再为其发送通知），s0283（静默存储事件直到取回）。	s0325, s0327, s0386
S04	更新后的订阅用于后续通知	更新之后，后续通知遵循更新后的订阅参数：订阅当前绑定的 notificationURI 和静默状态，不回退到被替换的早先上下文。	4.2.2.2.3 s0300..s0303：NWDAF 更新并存储订阅，NOTE 1 说 notificationURI 可被修改，以便后续通知发往新消费者。	s0298, s0300, s0303, s0306
K02	分析负载仅发送给其订阅者	事件通知中携带的分析信息，除订阅它的 NF 服务消费者和产生它的 NWDAF 之外，对其他任何一方都不可见，包括其他合法 NF。	5.1.9 将该 API 置于经 OAuth2 授权的 SBI 访问之下，TS 33.501 要求 SBI 信令机密性；通知体在 eventNotifications 中携带分析信息（s0343..s0367）。	ts-33-501:2871, s0343, s0340, s0565
Z03.10	令牌主体与订阅所有权	NWDAF 确信：某操作所用的 OAuth2 令牌是签发给出示该令牌的 NF 的，且其所操作的 subscriptionId 与 notificationURI 属于该 NF，因此任何令牌或订阅转让都不会把某项权利或某条通知转移给另一个消费者。	5.1.9 列出了授权类型、授权服务器和范围（s0565..s0570），但原文没有规定将订阅绑定到创建它的消费者的规则；RFC 6749 10.3 指出，OAuth2 本身没有为资源服务器提供方法，用来检查出示的访问令牌是否签发给出示令牌的客户端。	rfc6749:2879, rfc6749:2827, s0565, s0568, s0570, s0303
U01	通知不跨订阅	NF 服务消费者只接受属于自己的订阅的通知：通知中的 subscriptionId 和事件标识符属于它创建的订阅，且该通知不是为另一个消费者产生的。	s0383：通知携带事件 subscriptionId；s0340：通知发往该订阅绑定的 notificationURI；s0388..s0390：消费者存储通知并应答 204。	s0340, s0383, s0385, s0388
U02	更新或取消后无过期通知	更新或取消生效之后，NF 服务消费者不接受由旧订阅参数产生的通知；只有变更生效时已在途的通知才可被接受。	s0300..s0306（更新被存储、响应已发送）和 s0325..s0327（订阅被移除、204）说明了变更何时生效；文稿没有规定跨越该时刻的在途通知如何处理。	s0300, s0306, s0325, s0327, s0386

分析者没有补充性质：起草的清单已覆盖文稿写明的目标。

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 398 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 24 段，没选 374 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	rfc6749:2778-2786	没有陈述目标	引言部分指向 OAuth 威胁模型，本身没有提出任何要求。
P003 至 P006	rfc6749:2790-2823	Z01, Z03.10, `malicious_nf`	
P007 至 P010	rfc6749:2827-2850	Z03, Z03.10, `malicious_nf`	
P011 至 P014	rfc6749:2859-2877	Z01, K02, `network`, `malicious_nf`	
P015	rfc6749:2879-2881	Z03.10, Z03, `malicious_nf`	
P016 至 P021	rfc6749:2885-2917	不在范围内	刷新令牌；本 API 只用 Client Credentials 授权（s0565），该授权没有刷新令牌。
P022 至 P029	rfc6749:2921-2980	不在范围内	授权码授权的授权码和重定向 URI；本 API 不使用。
P030 至 P033	rfc6749:2984-3003	不在范围内	资源所有者口令凭据授权；本 API 不使用。
P034 至 P035	rfc6749:3013-3019	Z01, K02, `network`	
P036	rfc6749:3023-3028	D01, `network`	
P037 至 P039	rfc6749:3032-3041	证明器问不了	猜测客户端凭据取决于熵和速率限制，证明器无法计算。
P040 至 P062	rfc6749:3045-3210	不在范围内	钓鱼、跨站请求伪造、点击劫持、输入校验、开放重定向和隐式授权：这里的消费者是网络功能，不是浏览器或资源所有者，而且本 API 使用不带用户代理的 Client Credentials 授权。
P063	ts-33-501:2491-2491	不在范围内	UE 与网络之间对 5G 接入安全特性的降级协商，与 SBI 分析 API 无关。
P064 至 P074	ts-33-501:2495-2517	不在范围内	UE/SUPI 认证、服务网络授权和 5G 密钥长度；所选流程既没有 UE，也不涉及接入安全。
P075 至 P125	ts-33-501:2523-2631	不在范围内	对 UE、AS/NAS 机密性和完整性、签约凭据存储和签约用户隐私的要求；所选流程按非 UE 聚合分析来分析，任务也不涉及 UE 数据。
P126 至 P178	ts-33-501:2637-2761	不在范围内	对 5GC/NG-RAN 密钥、gNB、ng-eNB 和 F1/E1 的要求；与 SBI 分析 API 无关。

段落	出处	定为	不选的原因
P179 至 P198	ts-33-501:2767-2811	不在范围内	接入网（RRC/NAS）和 SEAF 上的信令保护；这些段落讲的不是本 API 的 SBI 信令。
P199 至 P213	ts-33-501:2819-2853	不在范围内	对 UDM、AUSF 和 UDR 签约数据的要求；用户同意数据不在所选流程内，该流程不携带 UE 标识符。
P214	ts-33-501:2859-2859	没有陈述目标	关于运营商信任域的假设，后续要求以此为基础。
P215 至 P217	ts-33-501:2865-2869	没有陈述目标	NF 发现和注册的机密性、完整性、重放保护和拓扑隐藏；本 API 是服务消费，不是发现。
P218 至 P220	ts-33-501:2871-2875	Z01, D01, A04, `network`, `malicious_nf`	
P221 至 P222	ts-33-501:2879-2881	没有陈述目标	描述 NRF 在发现和签发令牌中的作用。
P223 至 P227	ts-33-501:2883-2891	Z01, Z03.10, `malicious_nf`	
P228 至 P241	ts-33-501:2895-2923	不在范围内	NEF 要求和 SCP 要求；所选流程是一个消费者与一个 NWDAF 之间不经过 SCP 或 NEF 的直接通信。
P242 至 P254	ts-33-501:2929-2953	不在范围内	面向漫游中间方（IPX、SEPP）的核心网端到端互连安全；任务排除漫游交换。
P255 至 P314	ts-33-501:2957-3081	不在范围内	SEPP、N32-c/N32-f、跨 PLMN 的属性保护和 IPUPS；所选流程在一个 PLMN 之内。
P315 至 P318	ts-33-501:3085-3091	没有陈述目标	始发 PLMN/SNPN 头；这是网络间路由的建议，不是本 API 的性质。
P319 至 P337	ts-33-501:3097-3141	没有陈述目标	信令监控、安全可见性和运营商部署的可配置性。
P338 至 P369	ts-33-501:3149-3215	不在范围内	AS/NAS 保护的算法标识符和算法选择；模型把密码原语视为理想原语，任务也排除计算安全性和抗量子能力。
P370 至 P375	ts-33-501:3219-3231	不在范围内	5G-RG 和 NSSAAF 要求；与本 API 无关。
P376 至 P380	ts-33-501:6540-6548	不在范围内	N3 用户面接口上的安全。
P381 至 P394	ts-33-501:9656-10917	不在范围内	EAP-TLS 附着和 IAB 安全要求。

段落	出处	定为	不选的原因
P395 至 P398	ts-33-501:11439-11901	不在范围内	MBS 和联邦学习 NEF 要求；与本 API 无关。

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 22 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 15 项，没设的 7 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 service consumer	参与方	`malicious_nf`	
S02	NWDAF	参与方	按定义全部失守	这里每个目标都关于 NWDAF 的授权和通知行为；如果它不诚实，它所保证的性质按定义就不成立，而文稿在不诚实的 NWDAF 下也不要求其中任何一条。
S03	NRF	参与方	按定义全部失守	NRF 是授权服务器；它失守，所有授权性质按定义就都失去了。
S04	UDM	参与方	不在范围内	用户同意数据涉及面向 UE 的分析；所选流程是非 UE 聚合的网络性能，不携带 SUPI 或 GPSI。
S05	其他合法 NF	参与方	`malicious_nf`	
S06	其他 NF 消费者	参与方	`malicious_nf`	
S07	OAuth2 access token	值	`malicious_nf`	
S08	scope nnwdaf-eventssubscription	值	`malicious_nf`	
S09	scope nnwdaf-eventssubscription:transfer	值	不在范围内	nnwdaf-eventssubscription:transfer 范围涉及分析订阅转移，任务已将其排除。
S10	用户同意订阅数据	值	不在范围内	用户同意订阅数据通过 Nudm_SDM 获取；所选流程没有 UE，任务也不涉及 UDM 交互。
S11	subscriptionId	值	`malicious_nf`	
S12	notificationURI	值	`malicious_nf`	
S13	notifCorrId	值	攻击者得不到更多	通知关联标识符是通知里携带的不透明值；它不授权任何东西，只对一个消费者有意义。
S14	事件通知	值	`malicious_nf`	
S15	eventSubscriptions	值	`malicious_nf`	
S16	订阅资源	值	`malicious_nf`	
S17	订阅请求	值	`malicious_nf`	
S18	订阅状态	值	`malicious_nf`	
S19	通知	值	`malicious_nf`	
S20	notifFlag 状态	值	不在范围内	notifFlag 的静音和读取状态属于 EneNA 特性，所选流程没有对其建模。
S21	事件通知载荷	值	`malicious_nf`	
S22	分析数据	值	`malicious_nf`	

每条性质在各攻击者条件下的答案

攻击者由分析者设定，在写查询之前就已设定；每个答案都在某个攻击者下记录。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"是给该性质设定的、证明器在其下把主张留作未决或没有记录任何内容的攻击者；空单元格是分析者没有为该性质设定的攻击者。

攻击者	持有什么	依据	出处
`network`	只有网络：它能读取、丢弃、修改和发送消息，不持有任何一方的秘密		
`malicious_nf`	它自己的 NF 客户端凭据，以及 NRF 签发的、带有 nnwdaf-eventssubscription 范围的有效 OAuth2 访问令牌，因此它可以以自己的身份调用 Subscribe、Update、Unsubscribe 并应答通知；它可以保留、重放和重排收到的每条消息，还可能从自身角色暴露之处得知另一消费者的 subscriptionId 或 notificationURI。	任务范围把外部网络攻击者与持有自己凭据的恶意合法 NF 区分开来；5.1.9 把唯一的范围 nnwdaf-eventssubscription 授予每个已授权消费者（s0565..s0570），而 TS 33.501 5.9.2.1 只要求 NRF 授权该 NF，并不要求 NWDAF 把资源与其创建者绑定。	ts-33-501:2883, ts-33-501:2875, s0565, s0568, s0570

ID	`network`	`malicious_nf`
Z01	成立	成立
Z03	无答案	不成立
D01	成立	成立
A03	成立	不成立
A04	无答案	无答案
R02	无答案	无答案
S05	无答案	无答案
S04	无答案	无答案
K02	成立	成立
Z03.10	成立	不成立
U01	成立	成立
U02	无答案	无答案

原文和目标

出处：609 个原文单元，498330 字节，冻结在 source-delivery-01.json 中。 目标：11 个已批准（Z01, Z03, D01, A03, A04, R02, S05, S04, K02, U01, U02）；所问问题：3 个。 目标清单在事先按草稿获批，使用 --approve-targets。

宿主按其文档化默认值（任务 `decisions: auto`）所做的决定，供你核对：

- approve_package: 批准源文包 protocol-9d033e9fb7e29d6bc9a30e01de6b

本步骤自报：2 个请求（2 个失败），0.000 USD，输入 422144 个 token，输出 11079 个 token，286 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	原因
50	Z01、Z03、Z03.10、D01、A03、A04、R02、S05、S04、K02、U01、U02	plain_primitives	SBI（双向 TLS）按端点建模，使用理想签名和理想公钥加密，OAuth2 令牌建模为已签名的三元组。证明器将其视为完美实现，因此不主张任何计算安全性或抗量子能力。所有关于授权、通知完整性和负载保密性的答案都建立在这一理想化之上；除此之外，令牌和密钥与流程保持一致。

分析者的结语

在 model.pv 上回答了十二个目标。该模型涵盖 subscribe、notify 和 unsubscribe 流程：SBI 按端点保护（签名、向接收方加密、每个负载绑定到其接收方）；OAuth2 令牌由 NRF 签发并由 NWDAF 校验；一个诚实消费者，以及一个凭据被攻击者掌握的第二 NF。

以下结论经证明确立，每项都有对照查询表明该性质可能不成立：没有向同一 NF 签发的令牌，任何操作都不会被接受（Z01，对恶意 NF 攻击者同样成立；Z01 中与同意有关的那一半不适用，因为该流程是非 UE 聚合）；通知经过认证、绑定到订阅的消费者，并且对两类攻击者都保密（D01、K02、U01）；消费者的 201 是真实的（A03）；签发给一个 NF 的令牌不能被另一个 NF 使用（Z03.10）。

发现一个弱点，涉及 Z03、Z03.10 和 A03 的答案：NWDAF 没有将订阅绑定到创建它的消费者，因此另一个持有同一范围有效令牌的 NF 可以退订并非由它创建的 subscriptionId，NWDAF 还会删除指定消费者从未要求删除的订阅。原文未规定所有权规则（s0278、s0324..s0327），RFC 6749 也指出 OAuth2 无法让资源服务器检查令牌是签发给谁的。该发现在新颖性上标记为不明确，因为 TS 33.501 只要求每个 NF 校验传入消息。建议是：将令牌主体与资源一同记录，并在 Update 和 Unsubscribe 时进行比较。该建议已在模型的副本上检查：记录的攻击不再存在，正常流程仍能找到，其余十三个已记录的答案保持不变。

其余目标的阻碍：五个目标（A04、R02、S05、S04、U02）仅通过可达性回答，因为 ProVerif 的表只增不减，无法表示订阅的删除或替换、更新或取消生效的瞬间，也无法表示消息计数。下一步是构建有界模型，将消费者和 NWDAF 展开到三步，订阅状态沿链路传递，随后对重放、重复一次性操作、失同步和无回滚模式进行状态审查。notes.md 中记录了两条模型注意事项：模型公开了第二 NF 的密钥，因此每个“网络”答案实际上是对持有某一 NF 凭据的攻击者的答案；PUT 未建模，因此 notificationURI 重定向这一同样缺失绑定的问题依据 s0303 论证，而非展示。