Formal Verification of the FDO protocol

Simone Bussa, Riccardo Sisto, Fulvio Valenza

DAUIN, Politecnico di Torino, Italy

{first}.{last}@polito.it

Politecnico di Torino

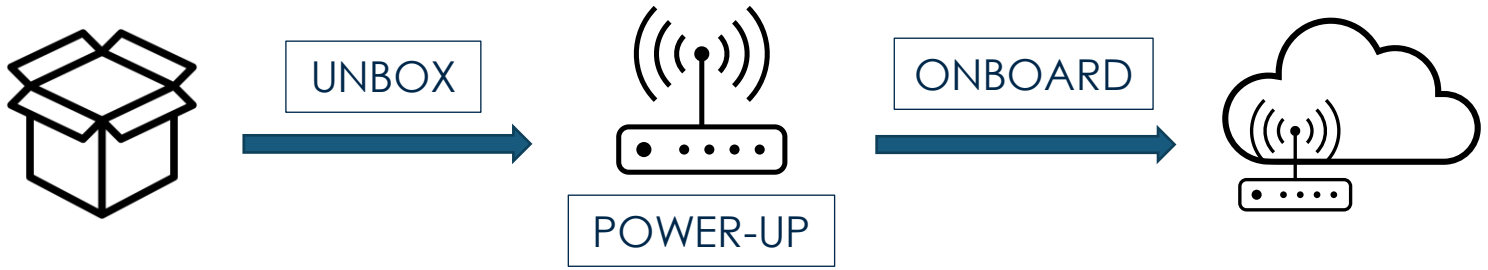
1859

netgroup

DAUIN

IoT Device Onboarding Introduction

IoT Device Onboarding



Onboarding solutions today

- Manual installation: time, trust, costs
- Proprietary ‘zero touch’ protocols: specific platforms, pre-configuration

Need to replace proprietary protocols with a single shared standard

Fido Device Onboarding

In 2020, Fido Device Onboarding (FDO)

- By the FIDO Alliance
- Hardware independent, plug&play
- Allows late binding

FDO v1.1 [1] is a Proposed Standard Specification

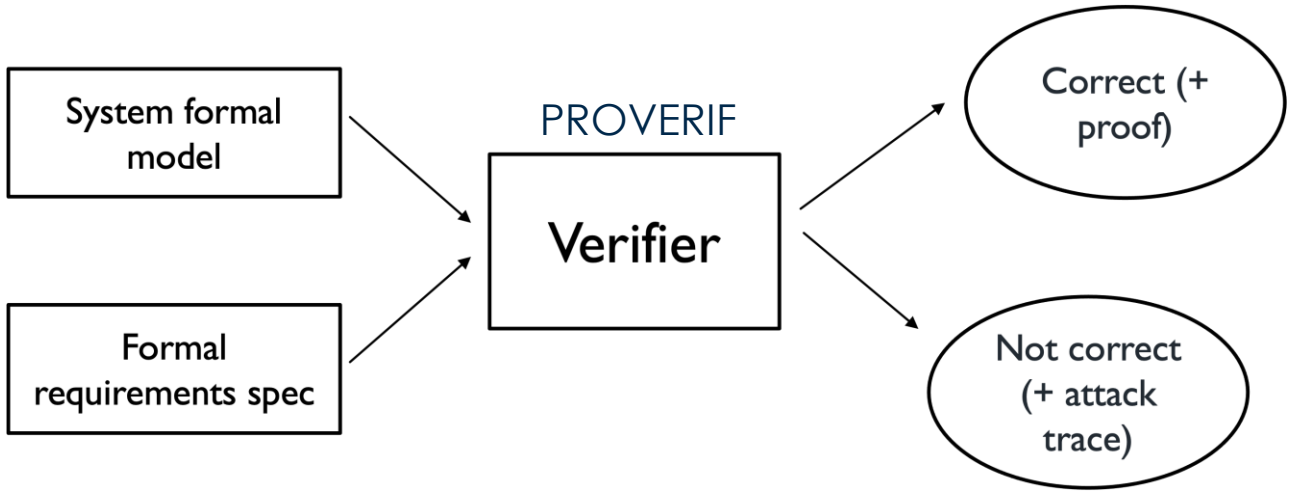
- It is necessary to deeply analyze its security
- No formal verification found in the literature

My contributions: perform a first formal analysis of the protocol to highlight potential vulnerabilities

Formal Verification

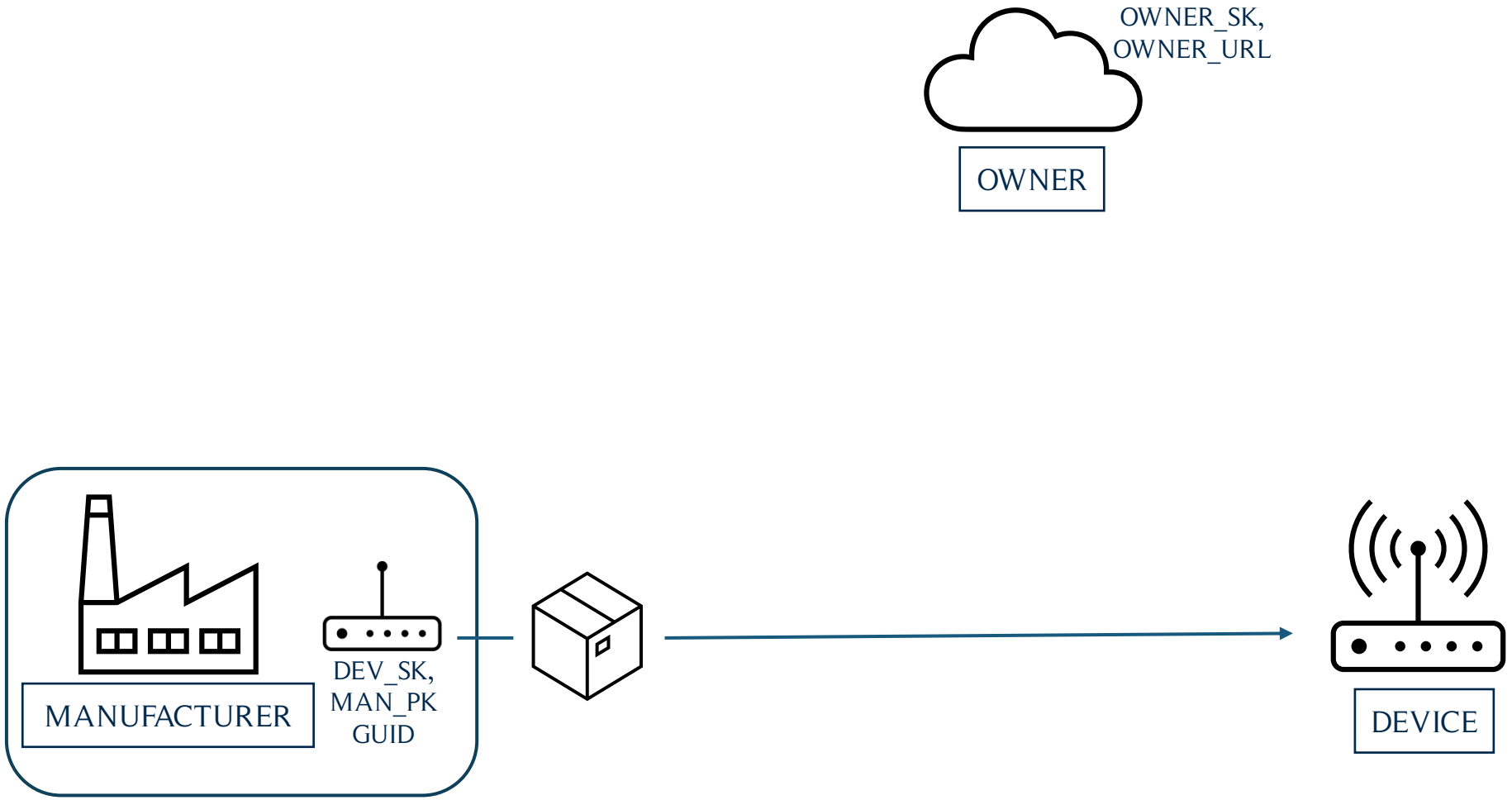
Static analysis of a system formal model

1. Formal Specification: from the system to its formal abstract model
2. Formal Verification: check if the model satisfies some formal properties

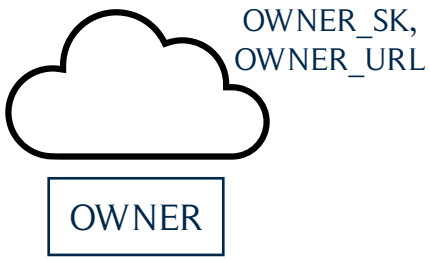


Formal methods increase the confidence in the security of the protocols by giving high assurance

FDO Specification

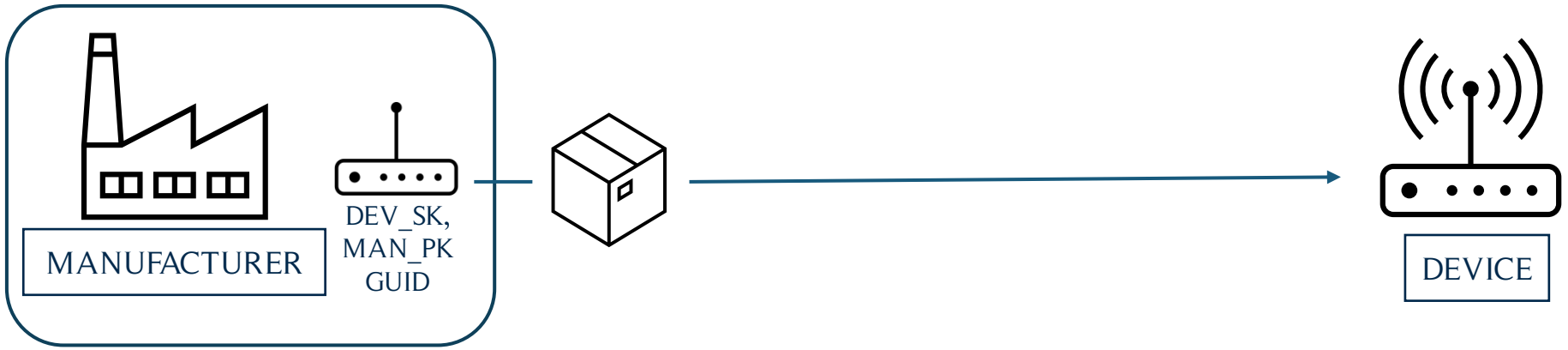


FDO Specification

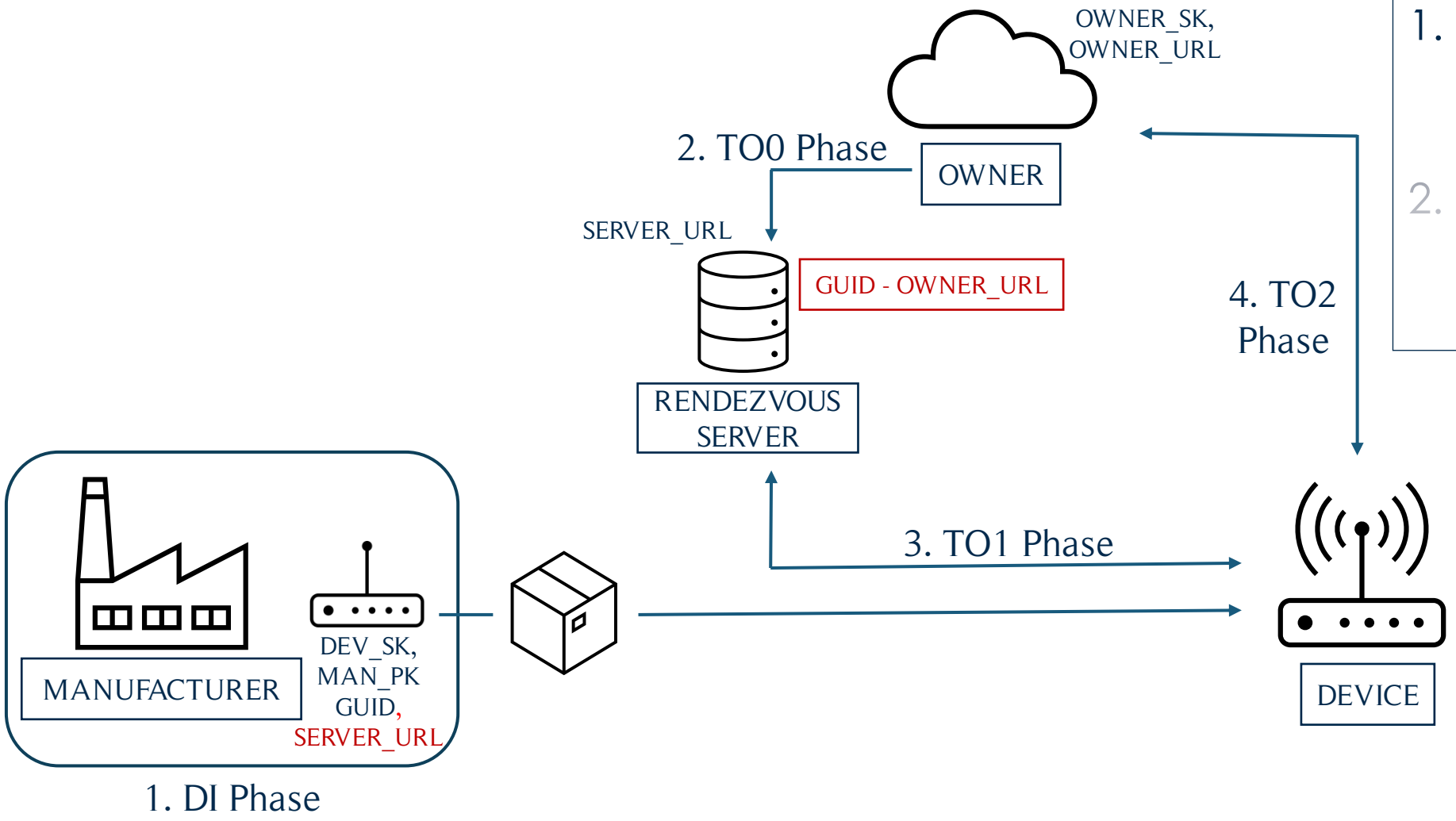


Two questions:

1. How does the device discover the Owner URL?
2. How is the transfer of ownership of the device handled?



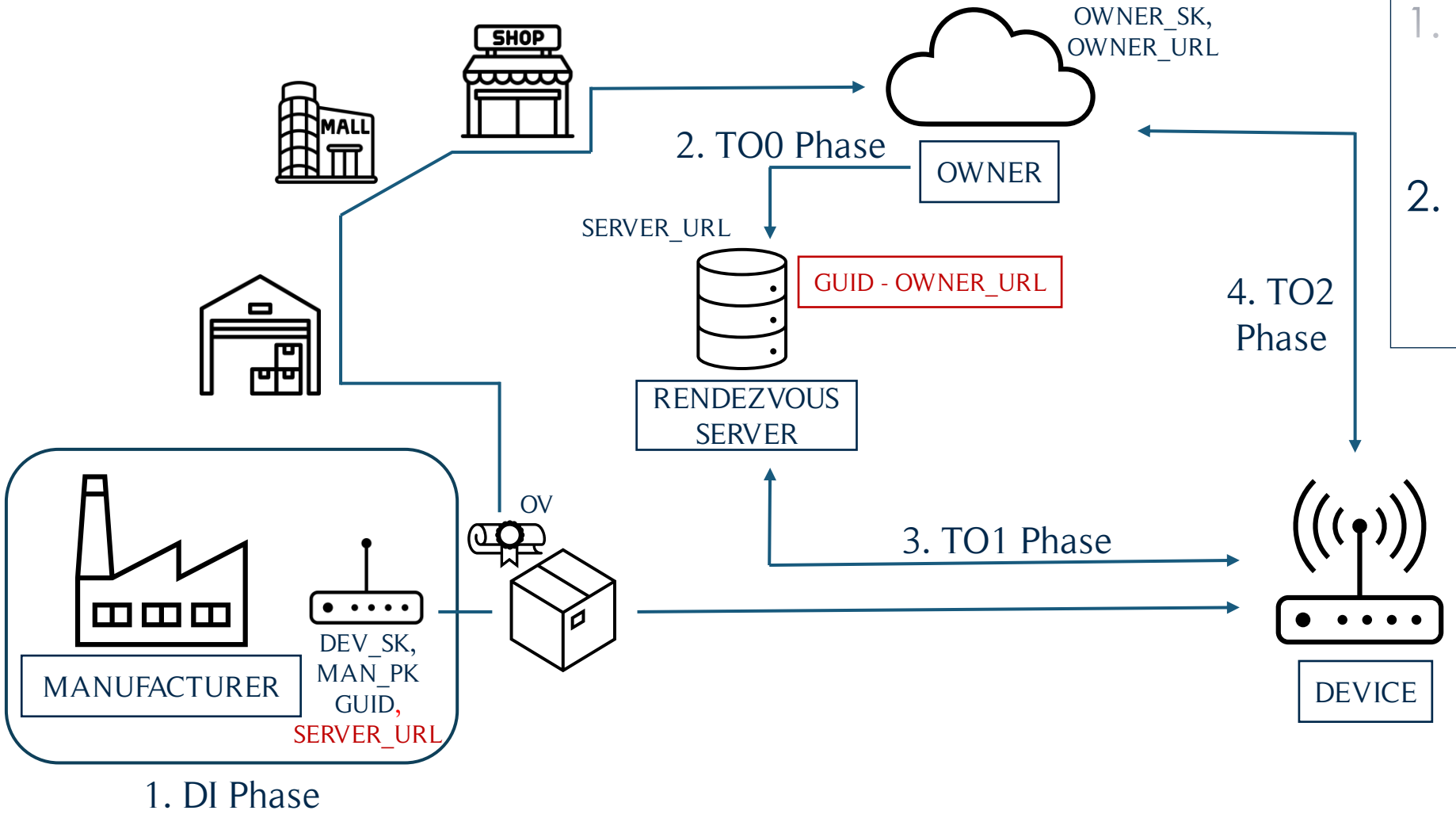
FDO Specification



Two questions:

1. How does the device discover the Owner URL?
2. How is the transfer of ownership of the device handled?

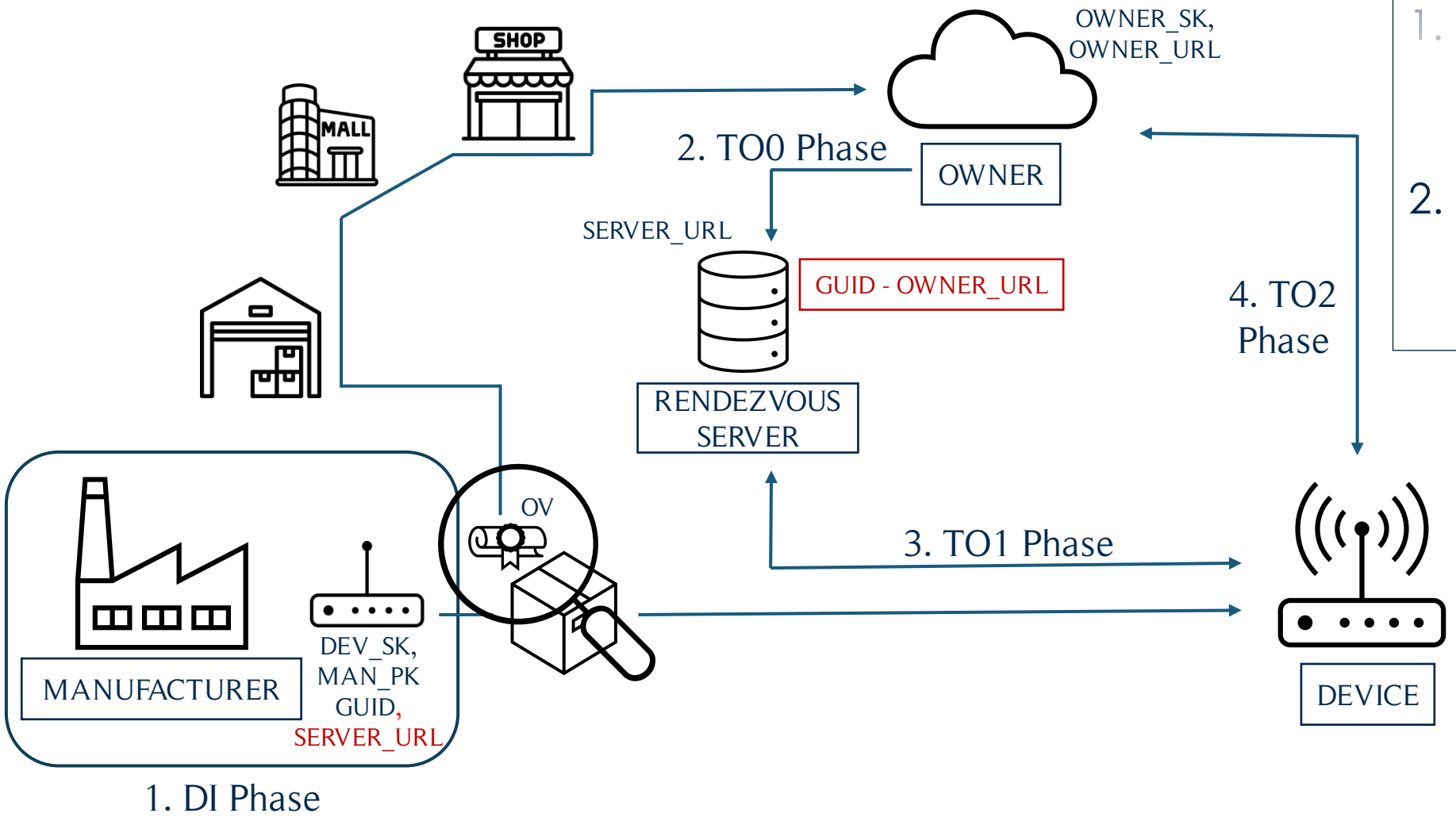
FDO Specification



Two questions:

1. How does the device discover the Owner URL?
2. How is the transfer of ownership of the device handled?

FDO Specification



Two questions:

1. How does the device discover the Owner URL?
2. How is the transfer of ownership of the device handled?

Ownership Voucher

OVHeader
GUID, SERVER_URL, MAN_PK, DEV_PK

OVEntryArray

MAN_PK,
h(OVHeader),
sign(MAN_SK)

Ownership Voucher

OVHeader
GUID, SERVER_URL, MAN_PK, DEV_PK

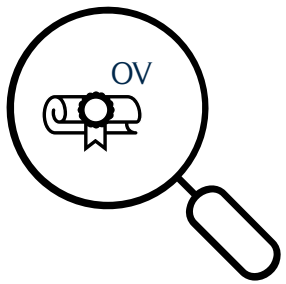
OVEntryArray

MAN_PK,
h(OVHeader),
sign(MAN_SK)

DIST_PK,
h(entry$_0$),
sign(MAN_SK)

Signed using private key corresponding to

Ownership Voucher



OVHeader
GUID, SERVER_URL, MAN_PK, DEV_PK

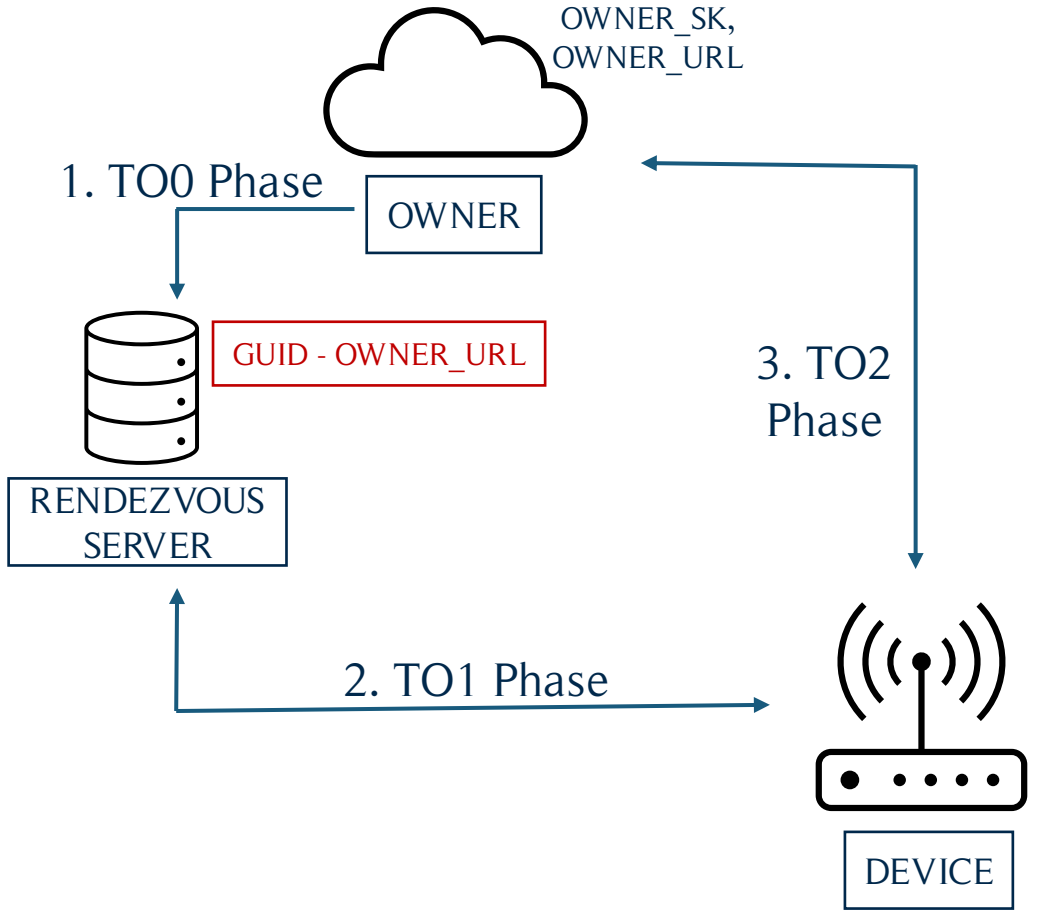
OVEntryArray

Owner has sk_N_i

MAN_PK, h(OVHeader), sign(MAN_SK)	DIST_PK, $h(entry_0)$, sign(MAN_SK)	...	pk_Ni, $h(entry_{i-1})$, $sign(sk\_N_{i-1})$

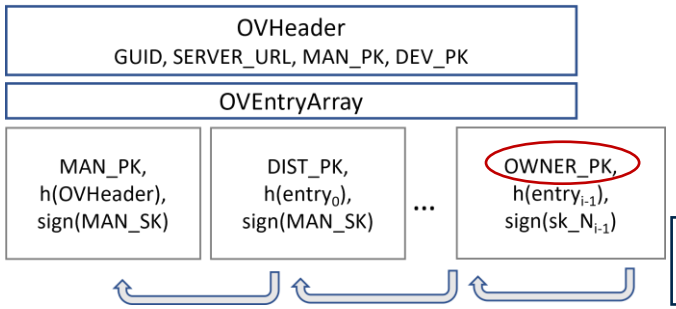
Signed using private key corresponding to

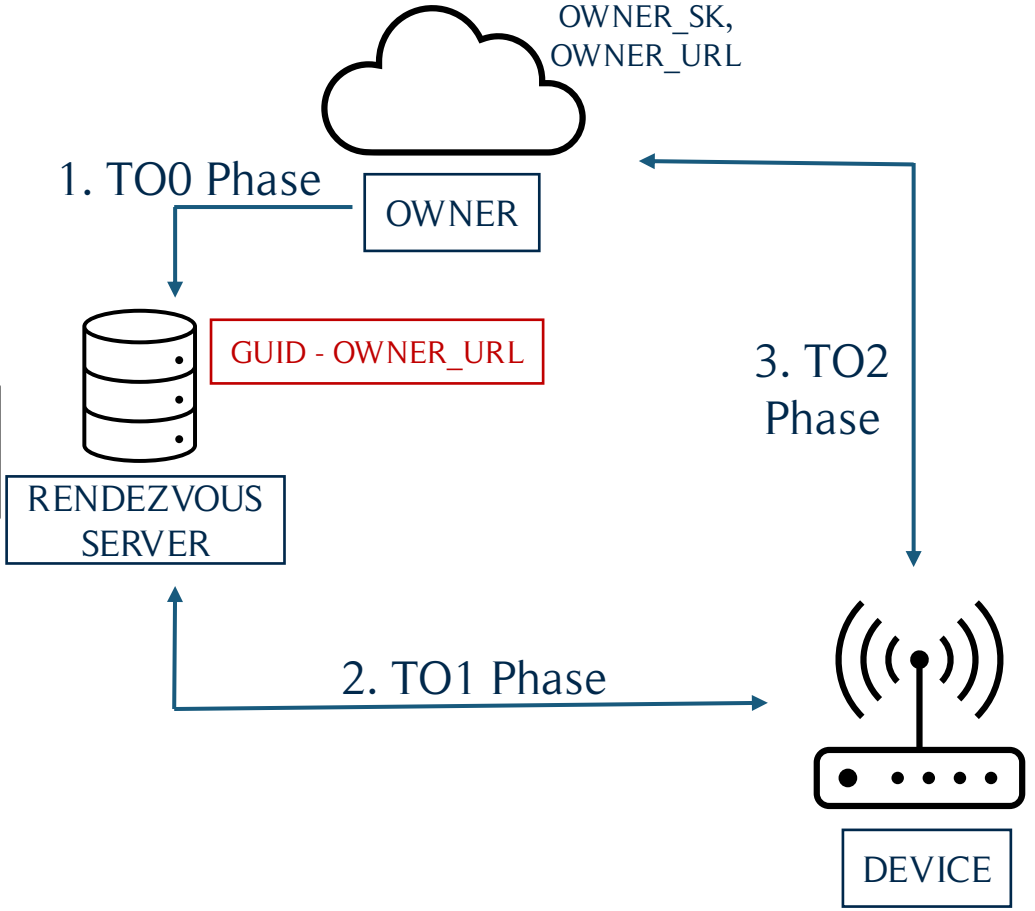
Authentication in the FDO protocol



Authentication in the FDO protocol

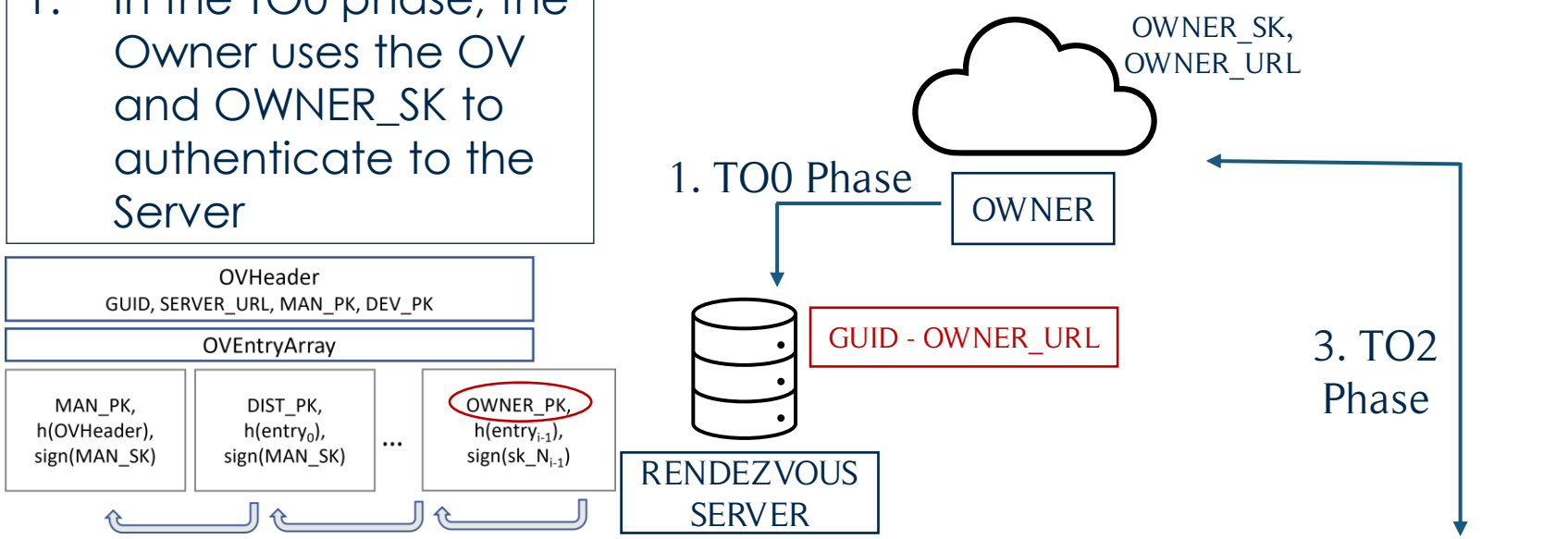
1. In the TO0 phase, the Owner uses the OV and OWNER_SK to authenticate to the Server





Authentication in the FDO protocol

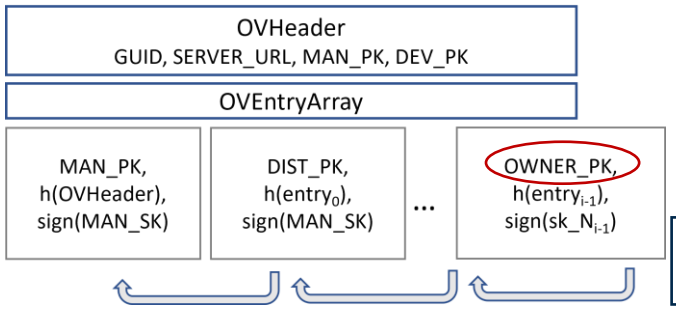
1. In the TO0 phase, the Owner uses the OV and OWNER_SK to authenticate to the Server

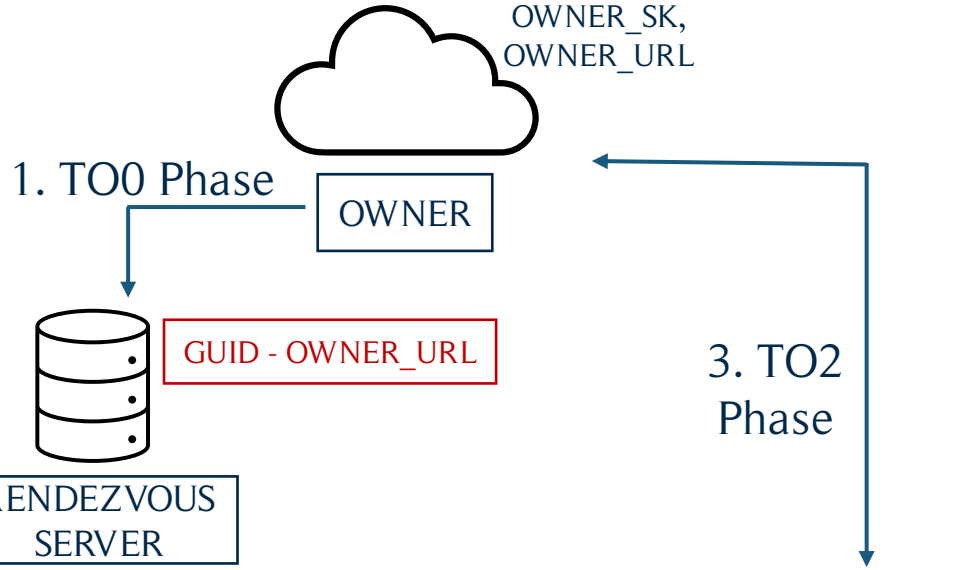


2. In the TO1 phase, the device uses DEV_SK to authenticate to the Server

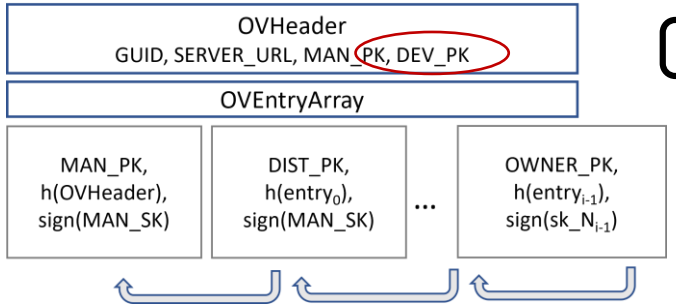
Authentication in the FDO protocol

1. In the TO0 phase, the Owner uses the OV and OWNER_SK to authenticate to the Server





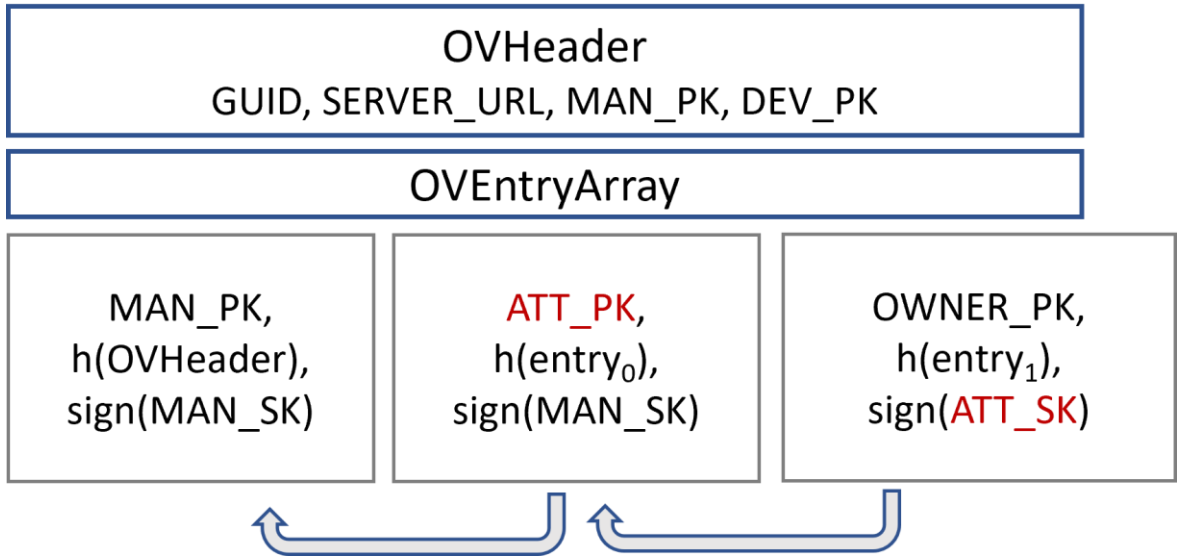
2. In the TO1 phase, the device uses DEV_SK to authenticate to the Server



3. In the TO2 Phase, the Owner uses the OV and OWNER_SK to authenticate to the device
4. While the device uses DEV_SK to authenticate to the Owner

Verification results

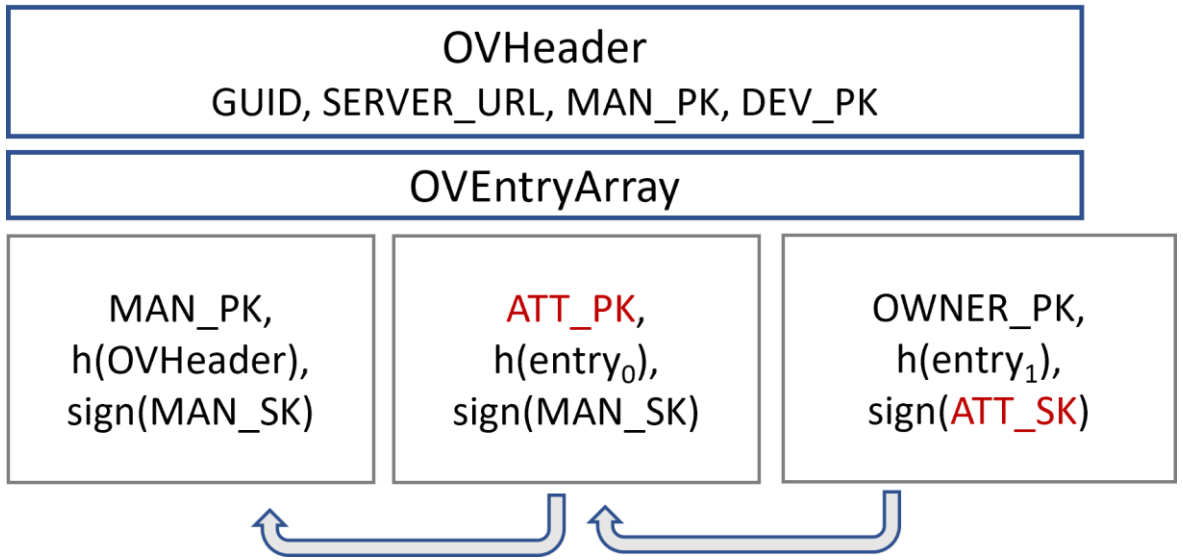
Proverif found the following weakness, when the attacker is an intermediate node in the supply chain



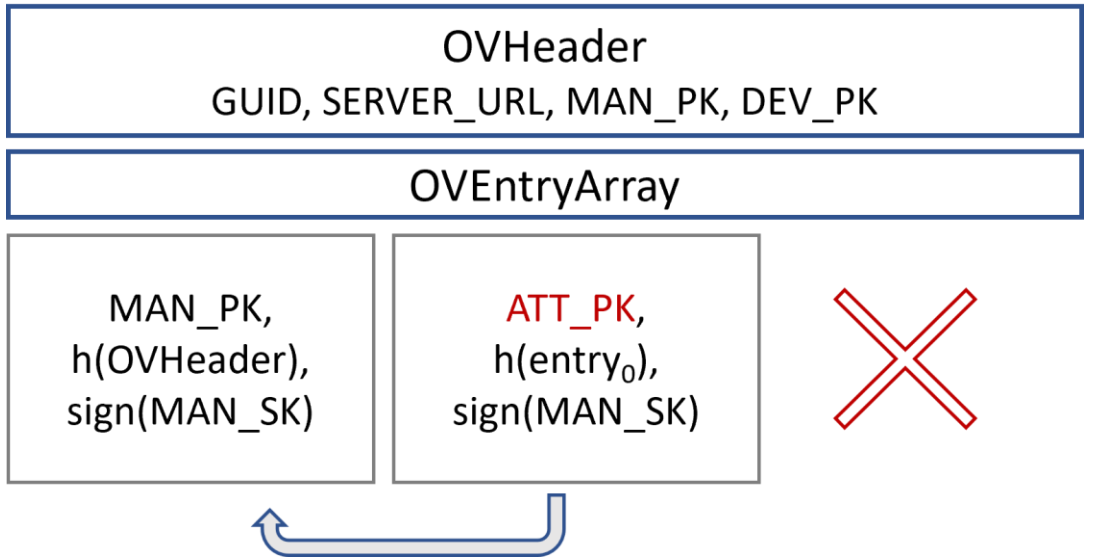
OV 3 entries authenticates the Owner

Verification results

Proverif found the following weakness, when the attacker is an intermediate node in the supply chain

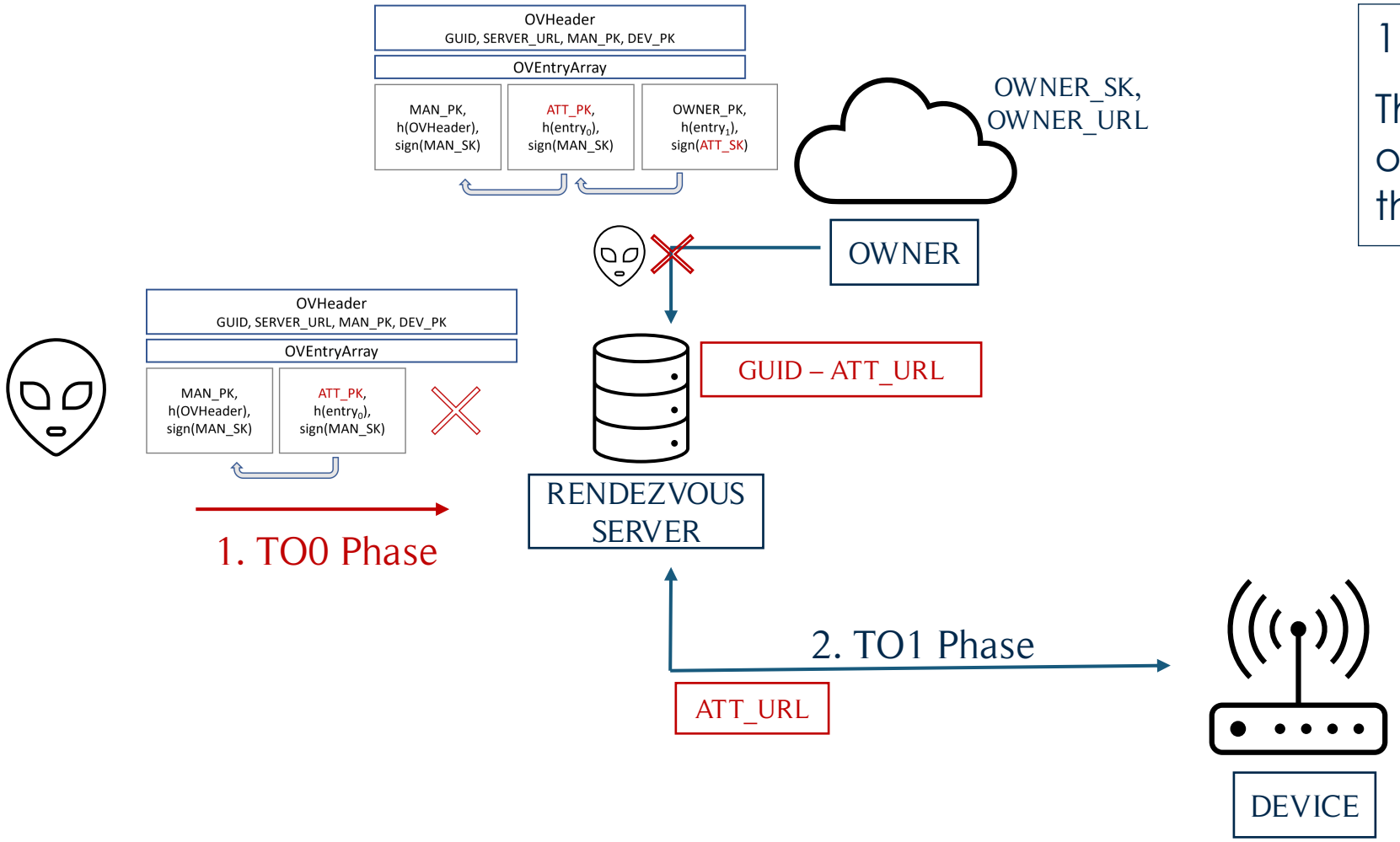


OV 3 entries authenticates the Owner



"Truncated" OV authenticates the attacker

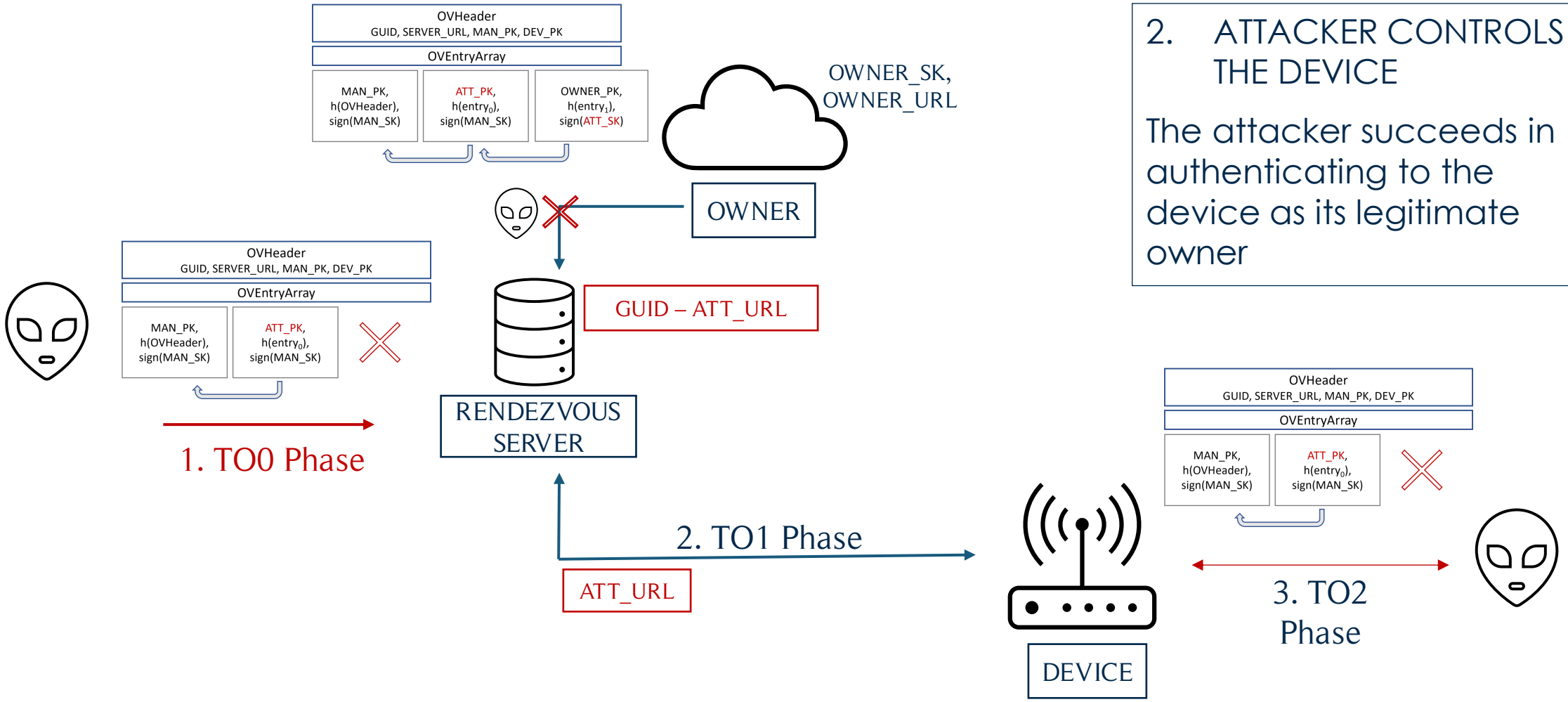
Consequences (I)



1. DENIAL OF SERVICE

The device will never open a connection with the real Owner

Consequences (II)

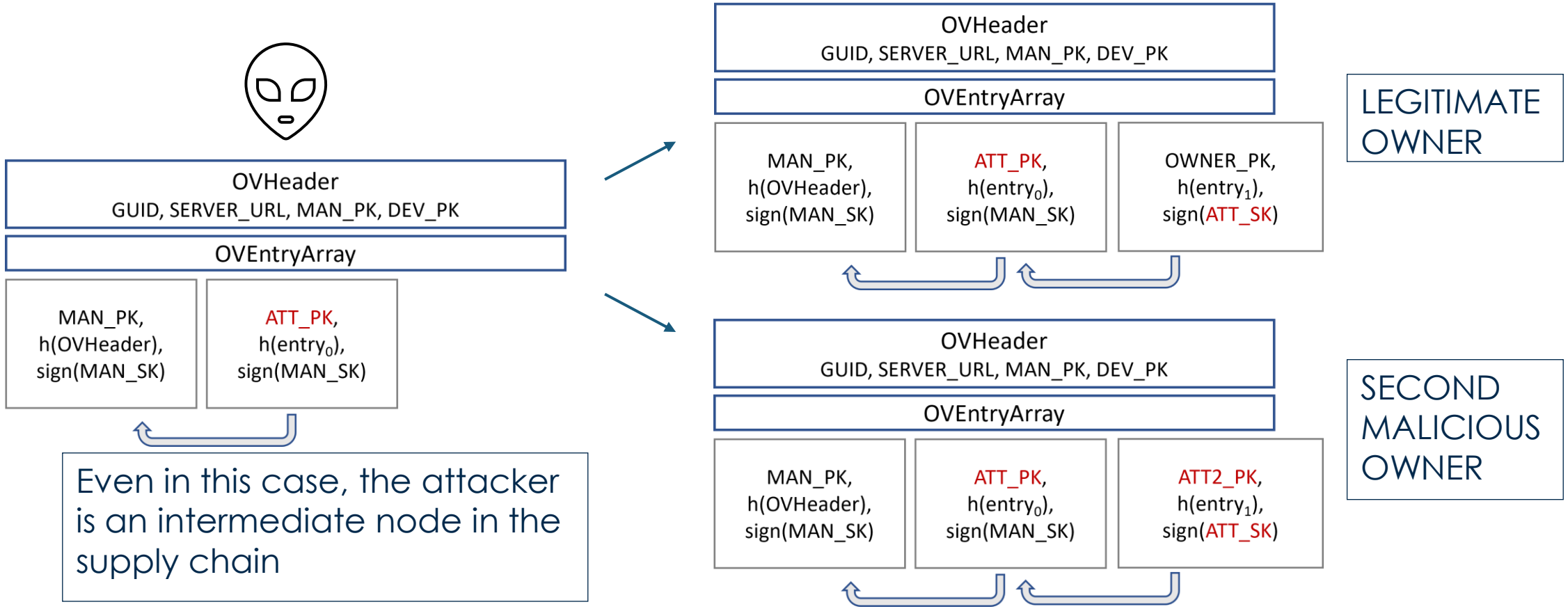


2. ATTACKER CONTROLS THE DEVICE

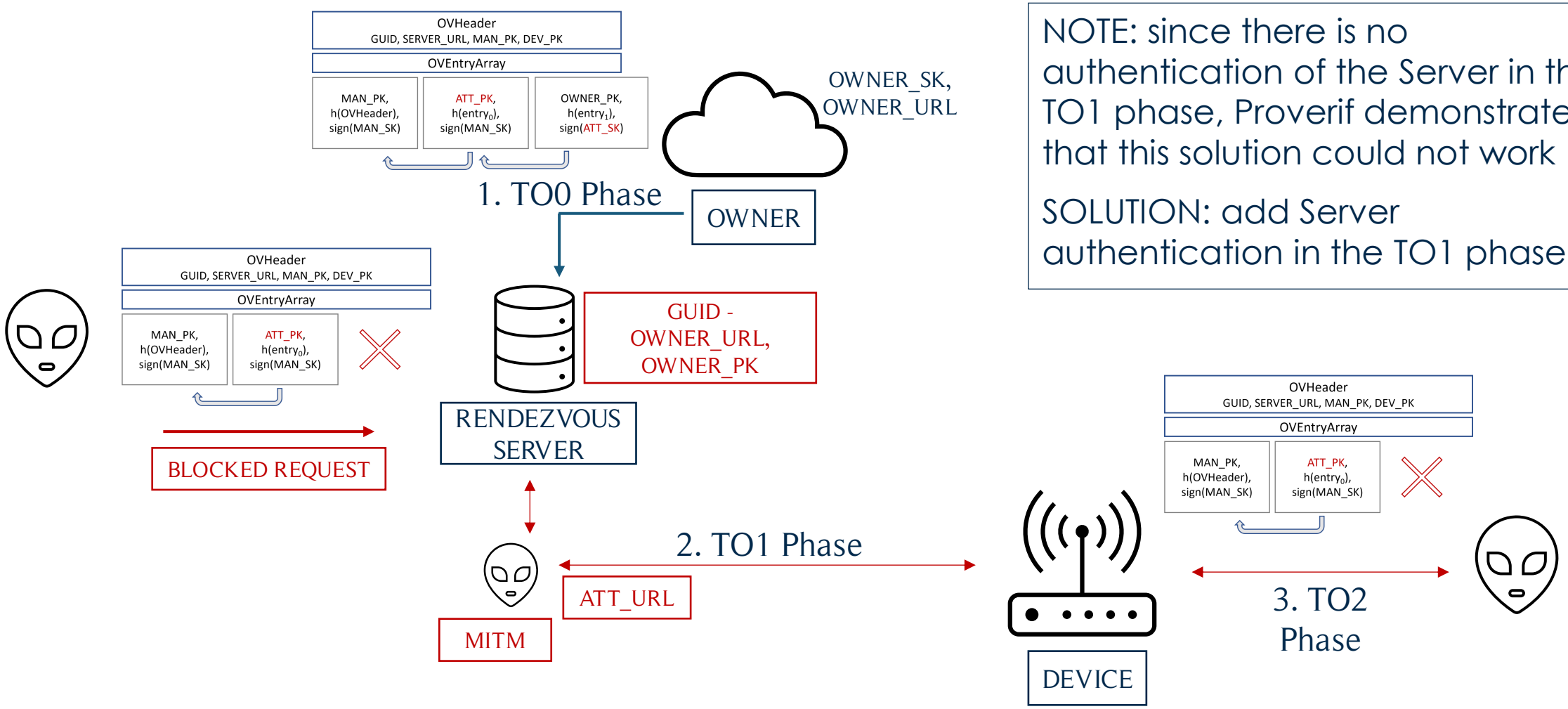
The attacker succeeds in authenticating to the device as its legitimate owner

Similar attack reported by FIDO

In an app-note FIDO reported a similar vulnerability



Possible countermeasures



NOTE: since there is no authentication of the Server in the TO1 phase, Proverif demonstrated that this solution could not work

SOLUTION: add Server authentication in the TO1 phase

Conclusions and Future work

Conclusions:

- First formal symbolic analysis of the FDO protocol
- Verification found a weakness similar to another one already known
- We reported our findings to FIDO who is conducting a stringent certification program to assess the security of the FDO protocol
- We hope our analysis can contribute to improve the protocol draft

Future work:

- Test the attack on the real implementation
- Propose countermeasures (Server authentication in TO1)

Thanks for your attention!

SIMONE BUSSA
simone.bussa@polito.it