Man-in-the-Middle Proof-of-Concept via Krontiris' Ephemeral Diffie-Hellman Over COSE (EDHOC) in C*

Daniel Hennig[†,‡] and Joaquin Garcia-Alfaro[†]

[†]SAMOVAR, Télécom SudParis, Institut Polytechnique de Paris, Palaiseau, France
[‡]INSA Toulouse, Département de génie electrique et informatique, France
E-mail: *hennig@insa-toulouse.fr,*
joaquin.garcia_alfaro@telecom-sudparis.eu

Abstract

This report presents part of the work carried out during a five-month research internship at the SAMOVAR laboratory of Télécom SudParis, focusing on some security aspects of B5G (Beyond 5G) networks. The internship combined literature review, protocol analysis, and simulation work. Particular attention was given to the authentication process of the Ephemeral Diffie-Hellman Over COSE (EDHOC) lightweight key exchange protocol, examining how Man-in-the-Middle (MitM) attacks could undermine trust models, e.g., under the scope of lawful interception and its risk to facilitate mass surveillance. We report only some technical aspects associated to the internship under the tasks associated to the aforementioned MitM attack scenario designed and implemented against the EDHOC protocol. Some other specific aspects of the work, mainly focusing on the security implications of malicious metasurfaces against B5G networks, are excluded from the scope of this report.

1 Introduction

The Ephemeral Diffie-Hellman Over COSE (EDHOC) protocol is a lightweight key exchange protocol for constrained devices (e.g., network nodes running on limited amount of memory space) or for scenarios under constrained network properties (e.g., in terms of bandwidth). It achieves the aforementioned goal by reducing the amount of messages sent, the message size, as well as by reusing primitives already used by other protocols [1, 2].

---

*This is a selected and revised version of Chapter 3 from Daniel Hennig's internship manuscript on 'Cybersecurity Impact of Malicious Reconfigurable Intelligent Surfaces', reproducing part of the work carried out during a five-month research internship (from April 2025 to September 2025) at the SAMOVAR laboratory of Télécom SudParis, within the activities of the SCN team.

As a result, EDHOC can run on IoT devices needing low energy consuming protocols as well as for Vehicle-to-Vehicle (V2V) or Vehicle-to-Infrastructure (V2I) structures. In the future, EDHOC could also be used as a B5G networks standard association protocol, for example between a Base Station and a User Equipment, to guarantee secure communications [1]. The basic version of the protocol is based on three mandatory messages as well as an optional fourth.

Figure 1 shows a sketch of the EDHOC message flow and their content. In the *first message*, from the Initiator (I) to the responder (R), I indicates the available methods and cryptographic suites as well as its Elliptic Curve Diffie-Hellman (ECDH) ephemeral public key (G_X) and a connection identifier (C_I) used to associate the public key to a specific connection, in the case of multiple connections at the same time on the same machine. In this first message, as well as in future messages, the EAD_n field is always used for External Authorization Data, for applications to directly
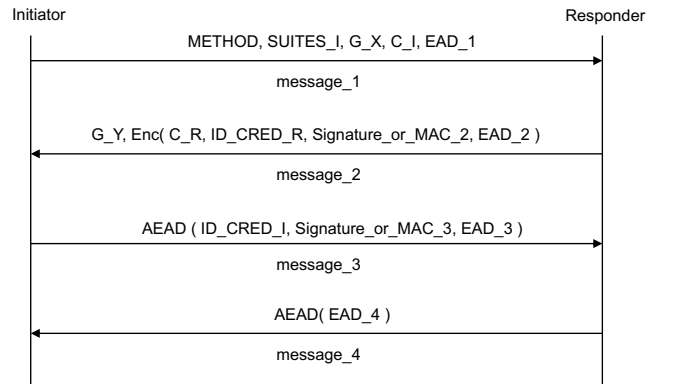

Figure 1. Extract of an EDHOC message exchange and their contents, as defined in [3].

Table 1. EDHOC authentication methods

Method type value	Initiator Authentication Key	Responder Authentication Key
0	Signature Key	Signature Key
1	Signature Key	Static DH Key
2	Static DH Key	Signature Key
3	Static DH Key	Static DH Key
4	Pre-Shared Key	Pre-Shared Key

put their security specifications into the EDHOC protocol. This field has a great impact on the size of the message and can be ignored by the Responder, respectively the Initiator, in the case where it is not specified as critical [3].

In the *second message*, R replies with its own public key (G_Y) and connection identifier (C_R) as well as with the ID_CRED_R and Signature_and_MAC fields used for authentication. The two last fields verify proof of possession of the private key in the different authentications methods, specified later. The second message is also composed of the next EAD field.

The *third message* should be encrypted as per Authenticated Encryption with Associated Data, meaning that it will encrypt its content with the key of the EDHOC protocol as well as with the data from the optional previous EAD fields. With this third message, authentication of the Initiator is performed. The message contains the ID_CRED_R as well as the Signature_and_MAC data of the Responder. The *fourth optional message* is used to strengthen authentication, especially in the case where static Diffie Hellman keys are used.

2 Authentication and Secrecy Properties

To authenticate the devices participating in the exchange, EDHOC has four original options, as well as a fifth one under review, presented in Table 1 (cf.[1, 3, 4] and citations thereof). These five authentication methods are relevant in the context of MitM attacks, since they may decide whether the attack would succeed or fail.

The basic version of the EDHOC protocol mentions weak resistance with respect to Post-Compromise Security (PCS) properties. PCS, often referred as well as backwards secrecy, refers to the capacity of a cryptographic protocol to *heal* back after corruption or compromise of long-term authentication credentials. It can also be defined as the capacity of a cryptographic protocol to bounce back after a given, finite interval, even when corruption of credentials happens, hence affecting the integrity of the communication channel between users [5].

Yet, a distinction is usually made between weak PCS and perfect PCS. The difference is made in the type of keys the adversary has access to. Weak PCS guarantees that once an adversary loses access to a party's secrets for one specific session, it can no longer decrypt future sessions. Perfect

PCS, on the other hand, would imply that even if long-term keys of a user have been compromised, the protocol is able to *heal* and restore confidentiality, i.e., making sure that no future sessions can be decrypted. Achieving perfect PCS is extremely demanding, since the idea persists that if an attacker has access to long-term keys of one user, it can then compute every same operation than the user to compute future session keys for each session. In practice, almost no lightweight protocol fully resists such attacks [6]. EDHOC, like many other modern key exchange mechanisms, provides forward secrecy but does not achieve perfect PCS, meaning that a compromise at a given moment of the long-term credentials has long-lasting consequences on the security of the channel.

3 Lawful Interception of Traffic

Results in [7] focus on EDHOC compliance with respect to Lawful Interception (LI), i.e., technical implementation of communication channels surveillance (e.g., Patriot Act in the US, granting federal authorities access to digital data owned by companies or private users, without their consent or notification). LI raises many questions, both societal and technical. If an authority has access to supposedly encrypted data, is this data truly protected, and could other actors gain access to it as well? More broadly, if an authority can decrypt user data at will, can that data still be considered secure? Ultimately, this is a question of trust: depending on the state in which a user lives, to what extent can authorities be considered entirely trustworthy?

Current research on LI focuses on enabling authorized authorities to access compromising data without entirely breaking end-to-end encryption. At the same time, the goal is to avoid drifting into mass surveillance, which can be defined as a situation where anyone can be subject to arbitrary interference in their privacy. Correct use of LI should be limited to non-arbitrary cases, and ideally to cases where the user has explicitly given consent.

In modern proposals for LI, the relevant problem is how to make the session key available to a third party, the lawful interceptor, without violating the principle of end-to-end encryption. The most common approach today involves key escrow [8], but this technique poses a fundamental problem: it requires the third party to have direct access to the encryption key at any time, which is essentially incompatible

with the idea of end-to-end protection. Work in [7] instead presents alternative methods, for example using three different keys: two from the users and one from the authorities. In this case, the authority must request the users' cooperation to obtain their keys before decryption can occur.

4 LI-Compliance for EDHOC

While the protocol is still in the standardization process [3], the LAKE working group of the IETF has already explored how to implement LI-compliance in EDHOC [7]. This is a particularly interesting use case, since it attempts to incorporate LI while still avoiding mass surveillance scenarios. In their proposal, the Initiator and Responder perform a standard EDHOC exchange without modification, while at the same time each protocol message is sent to a proxy and encrypted with the public keys of the participants and the authorities. To perform interception and recover the shared secret of EDHOC, all parties –including the Initiator, the Responder, and the authorized authorities – must contribute their respective secrets derived from their private keys.

This approach has several advantages. The endpoints run a standard EDHOC exchange, preserving compatibility and avoiding disruption of existing implementations. Interception is fine-grained, limited to the targeted session rather than enabling bulk surveillance. The requirement that all parties provide their share of the secret ensures that no single actor can unilaterally recover the key, mitigating the risk of indiscriminate monitoring. The scheme also maintains EDHOC's identity-protection guarantees, making it impossible to falsely attribute participation in a session to an innocent party.

Nevertheless, important limitations remain. Additional cryptographic operations, such as secret encapsulation and proofs of knowledge, add computational overhead, which may be prohibitive for constrained IoT devices. Furthermore, interception depends on the cooperation of all designated authorities: if even one refuses or is unavailable, interception cannot occur. The reliance on proxies and supporting infrastructure also creates residual trust issues, since any compromise of these entities could endanger user privacy. Finally, as the proposal is still under discussion, questions remain about its scalability and feasibility for large-scale deployment. More broadly, while the design reduces the risk of blanket surveillance, it nevertheless introduces a form of backdoor.

5 Implementation of the Attack

5.1 Assumptions

Our attack scenario assumes compromised long-term authentication credentials of the parties or key escrow techniques (e.g., known pre-shared cryptographic keys held in

trust, due to legally mandated situations). The same attack assumption would break any other centralized PKI-based authentication protocol.

5.2 Attack Scenario

We assume a malicious device able to act simultaneously as a malicious rely between two nodes. The goal would be to make an endpoint believe that it is communicating with a legitimate node and a legitimate node believe that it is communicating with an endpoint, similar to how a MitM attack between an end point and a base station would work. We can also consider a setting in which the malicious device has been configured to facilitate LI, with authorities treated as potential adversaries. The aim is not to defeat end-to-end cryptography, but to demonstrate how programmable propagation can make interception easier, more reliable, and less visible. In the end, we assume three endpoints, two acting as legitimate nodes and one acting as the malicious interception point.

Our scenario gets inspiration on the use of malicious metasurfaces conducting the attack presented in [9], referred to as Metasurface-in-the-Middle (MSitM) attack. The technical part of the attack is inspired from the CoopeRIS simulation framework [10], which models metasurface-assisted vehicular communications over OMNeT++ [11]. More precisely, CoopeRIS provides the specific case of Reconfigurable Intelligent Surface (RIS), as an emerging solution in B5G scenarios, and composed of electromagnetic elements capable of altering the phase of incident of radio waves, allowing them to actively shape and redirect the wireless environment in real time [12, 13, 14]. Building our scenario over CoopeRIS, several attack strategies were designed and implemented, including malicious traffic redirection, side-lobe eavesdropping, RIS-based MitM attacks, and RIS-assisted lawful interception. While the complexity of the framework limited some aspects of physical-layer modeling, these simulations provided valuable insights into how RIS may reshape both attack surfaces and defensive strategies in future wireless environments.

One metasurface is configured to illuminate the proxy consistently, while the other serves the legitimate receiver. This way, communications are both delivered and mirrored without degrading user-perceived quality. In such a deployment, users may remain unaware that traffic is being duplicated, because the *extra* path is realized in the physical layer rather than by inserting an active relay in the logical network path. This configuration gives a visual representation and simulation of the scenario, to get a grasp of how it could work. Although a RIS cannot decrypt robust end-to-end sessions such as those established with modern EDHOC authentication, it could serve as a way to facilitate global capture of data for interception, for example if there is only one proxy situated in a very large city for the sake of being more discrete, multiple RIS having a benign purpose ini-

tially could be used simultaneously to send all of their traffic towards the proxy as well. Figures 2 and 3 depicts the general idea of our RIS-assisted lawful interception attack.
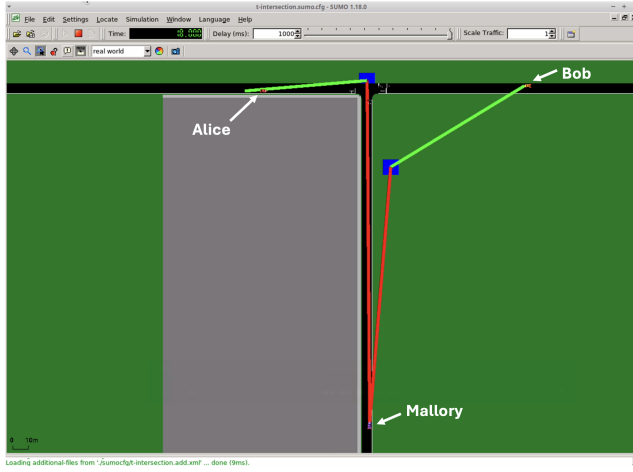

Figure 2. RIS-assisted lawful interception attack scenario (cf. companion github repository [15] for further details and videocaptures associated to the attack).
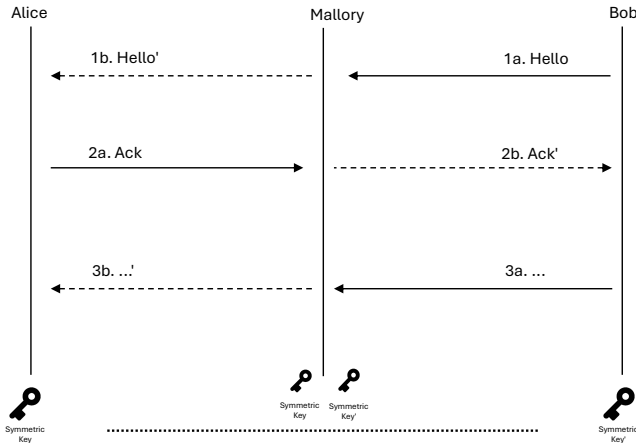

Figure 3. Sample representation of the attack w.r.t. the intercepted and modified message exchange flows.

The implementation associated to Figures 2 and 3 reuse existing code by Krontiris in [16], which provides a solid base of the EDHOC protocol in C language, which we modified under the context of the CoopeRIS framework [10] to simulate the attack PoC. The code was modified to ensure weak authentication between nodes, as defined in our assumptions (cf. Subsection 5.1). The full code, together with some videocaptures showing the most results of the attack,

can be found at our companion github repository[1] to foster further research on the topic.

6 Perspectives for future work

The issues reported in this document may raise the following research questions: how can endpoints or higher network layers reliably identify whether traffic is being redirected through malicious entities? One possible approach might involve the use of anomaly detection at the physical layer, e.g., by comparing expected propagation models with measured signal statistics. Another potential solution is the use of encryption over anamorphic channels [17, 18, 19], which could allow ciphertexts being decrypted into different messages with some sort of out-of-band verification (i.e., by concealing messages from the real content to the eyes of the surveillance framework). Some other practical solutions could rely on the use of decentralized Public Key Infrastructure (PKI) schemes, ensuring the use of different authorization platforms.

References

[1] Mališa Vučinić, Göran Selander, John Preuss Mattsson, and Thomas Watteyne. Lightweight Authenticated Key Exchange With EDHOC. *Computer*, 55(4):94–100, 2022.

[2] Elsa López Pérez, Göran Selander, John Preuß Mattsson, Thomas Watteyne, and Mališa Vučinić. EDHOC Is a New Security Handshake Standard: An Overview of Security Analysis. *Computer*, 57(9):101–110, 2024.

[3] Göran Selander, John Preuß Mattsson, and Francesca Palombini. Ephemeral Diffie-Hellman Over COSE (EDHOC). RFC 9528, March 2024.

[4] Elsa Lopez-Perez, Göran Selander, John Preuß Mattsson, and Rafael Marin-Lopez. EDHOC Authenticated with Pre-Shared Keys (PSK). Internet-Draft draft-ietf-lake-edhoc-psk-04, Internet Engineering Task Force, July 2025. Work in Progress.

[5] Olivier Blazy, Ioana Boureanu, Pascal Lafourcade, Cristina Onete, and Léo Robert. How fast do you

---

[1] Available online, cf. [15]

heal? a taxonomy for post-compromise security in secure-channel establishment. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5917–5934, Anaheim, CA, August 2023. USENIX Association.

[6] Cas Cremers, Niklas Medinger, and Aurora Naska. Impossibility results for post-compromise security in real-world communication systems. In *2025 IEEE Symposium on Security & Privacy (SP)*, pages 4391–4405, 2025.

[7] Pascal Lafourcade, Elsa Lopez Perez, Charles Olivier-Anclin, Cristina Onete, Clément Papon, and Mališa Vučinić. Fine-grained, privacy-augmenting LI-compliance in the LAKE standard - Extended version. In *Lecture Notes in Computer Science LNCS*, Toulouse, France, September 2025. Springer.

[8] Giuseppe Ungaro, Francesco Ricchitelli, Ingrid Huso, Giuseppe Piro, and Gennaro Boggia. Design and Implementation of a Lawful Interception Architecture for B5G Systems Based on Key Escrow. In *2022 IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 207–207, 2022.

[9] Zhambyl Shaikhanov, Fahid Hassan, Hichem Guerboukha, Daniel Mittleman, and Edward Knightly. Metasurface-in-the-middle attack: From theory to experiment. In *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 257–267, 2022.

[10] Michele Segata, Paolo Casari, Marios Lestas, Alexandros Papadopoulos, Dimitrios Tyrovolas, Taqwa Saeed, George K. Karagiannidis, and Christos Liaskos. CoopeRIS: A framework for the simulation of reconfigurable intelligent surfaces in cooperative driving environments. *Elsevier Computer Networks, Special Issue on Advances in Wireless Networks Simulation*, 248, 6 2024.

[11] Andras Varga. Omnet++. In *Modeling and tools for network simulation*, pages 35–59. Springer, 2010.

[12] Ertugrul Basar, Marco Di Renzo, Julien De Rosny, Merouane Debbah, Mohamed-Slim Alouini, and Rui Zhang. Wireless communications through reconfigurable intelligent surfaces. *IEEE Access*, 7:116753–116773, 2019.

[13] M. Di Renzo et al. Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead. *IEEE Journal on Selected Areas in Communications*, 38:2450–2525, 2020.

[14] Yuanwei Liu, Xiao Liu, Xidong Mu, Tianwei Hou, Jiaqi Xu, Marco Di Renzo, and Naofal Al-Dhahir. Reconfigurable intelligent surfaces: Principles and opportunities. *IEEE Communications Surveys and Tutorials*, 23(3):1546–1577, 2021.

[15] Daniel Hennig and Joaquin Garcia-Alfaro. Implementation of a Man-in-the-Middle attack on Krontiris' Ephemeral Diffie-Hellman Over COSE (EDHOC). `https://github.com/danielhng/EDHOC-C_MitM`, 2025.

[16] Alex Krontiris. Implementation of Ephemeral Diffie-Hellman Over COSE (EDHOC) in C. `https://github.com/alexkrontiris/EDHOC-C`, 2017.

[17] Giuseppe Persiano, Duong Hieu Phan, and Moti Yung. Anamorphic encryption: private communication against a dictator. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 34–63. Springer, 2022.

[18] Fabio Banfi, Konstantin Gegier, Martin Hirt, Ueli Maurer, and Guilherme Rito. Anamorphic encryption, revisited. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 3–32. Springer, 2024.

[19] Adrian Cinal, Przemysław Kubiak, Mirosław Kutyłowski and Gabriel Wechta. Anamorphic monero transactions: the threat of bypassing anti-money laundering laws. In *30th European Symposium on Research in Computer Security (ESORICS 2025)*. Springer, 2025.