POSTER: Using Verifiable Credentials for authentication of UAVs in logistics*

Ken Watanabe[1] and Kazue Sako[1]

Department of Computer Scinence and Communication Engineering,
Waseda University, 3-4-1 Okubo, Shinjuku-ku, Tokyo, Japan

Abstract. Verifiable Credentials (VCs) have been widely adopted as a format for asserting information through use of digital signature technique, and offer versatile usages. The World Wide Web Consortium (W3C) has standardized the data model for VCs [1], and the motivation for this study is to verify the applicability of its data model through a concrete, realistic scenario. For verification, we have chosen a scenario where Unmanned Aerial Vehicles (UAVs) collaborate to deliver packages. As a contribution of this study, we demonstrate how we can selectively disclose attributes of UAVs yet ensure its mission under an authorized contract. Furthermore, we propose a method for issuing receipts through use of VCs and Verifiable Presentations (VPs) that guarantees completion of delivery while UAVs remain anonymous. For this function, we developed a new format for VP, namely 'VP with Message.' These features have been implemented through a demo application.

Keywords: Verifiable Credential · Selective Disclosure · Authentication Protocol · UAV.

1 Introduction

A Verifiable Credential (VC), whose data model has been standardized at the World Wide Web Consortium (W3C)[1], is a credential that certifies a holder's set of attributes by the issuer's signature. The credential contains information about a holder's attributes such as names and identifiers. The holder can selectively disclose the attributes to the verifier without revealing unnecessary or sensitive data as a Verifiable Presentation (VP). The data model for VCs is used in many cases, such as COVID-19 vaccination certificates [2].

The motivation for our research is to verify the applicability of its data model in a concrete, realistic scenario that goes beyond just showing and checking a document like a vaccine certificate, and also involves mutual authentication. For verification we have chosen a scenario where Unmanned Aerial Vehicles (UAVs) collaborate to deliver packages.

In this study, we make several contributions. We demonstrate how to perform selective disclosure for privacy enhancement using VCs and VPs in this scenario.

* These research results were obtained from the commissioned research (No.03901) by National Institute of Information and Communications Technology (NICT) , Japan.

Furthermore, we propose a method for issuing receipts through use of VCs and VPs that guarantees completion of delivery while UAVs remain anonymous. Also, we found signed documents such as contracts can be expressed as a VC and serve as a source for mutual authentication. These features we propose have been implemented through a demo application.

2 A Scenario for Confirming the Applicability of VC

In this section, we present a scenario of UAV collaboratively delivering packages to check applicability of VCs. We describe players and basic procedures.

2.1 Players and Purpose

There are 5 types of players: *Sender* (S), *PrimeContractor* (PC), *Subcontractor* (SC), *UAV* (U) and *Recipient* (R). *Sender* asks *PrimeContractor* to deliver packages to *Recipient*. The *PrimeContractor* requests $Subcontractor_1$ and $Subcontractor_2$ to collaboratively deliver the packages. Each *Subcontractor* assigns the task to their respective UAVs. When relaying the packages, both players need to authenticate that they are working under same *PrimeContractor*. Also, a player that handed over the packages need a receipt from a *UAV* to confirm successful relaying. Each player has a unique identifier and a key pair.

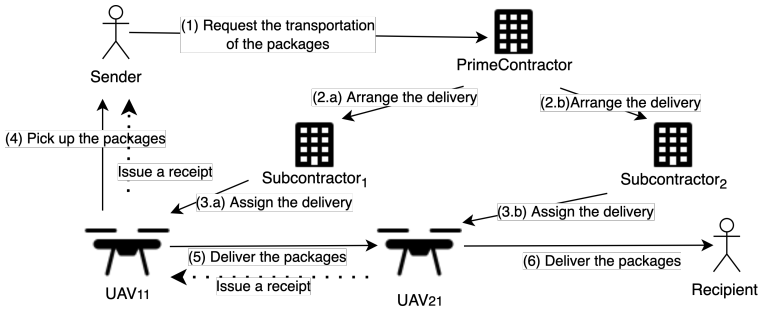

Fig. 1. Overall delivery scenario

2.2 Procedures in the scenario

Procedures in the scenario are depicted in Figure 1.

(1) *Sender* requests package delivery to *PrimeContractor*. (2.a, 2.b) *PrimeContractor* subcontracts the package delivery to $Subcontractor_1$ and $Subcontractor_2$. (3.a, 3.b) Each *Subcontractor* assigns the package delivery task to their respective UAVs, UAV_{11} and UAV_{21}. (4) UAV_{11} proceeds to pick up

the packages from *Sender* and (5) delivers it to the location of UAV_{21}. (6) Finally, UAV_{21} delivers the packages to *Recipient*. When relaying packages, the player who receive the packages issues a receipt.

3 Our system

In our system, we represent contracts as VCs. There are 4 types of contracts as detailed in 3.1. How to mutually issue VCs is explained in 3.2. The existence of contracts are proved through VP in selective-disclosure manner as described in 3.3.

3.1 Types of Verifiable Credential

Contracts using VC are utilized between the *Sender* and *PrimeContractor*, *PrimeContractor* and *Subcontractor*, and *Subcontractor* and UAV. The types of VC are shown in Table 1.

Table 1. 4 types of VC

VC	(Issuer, Holder)	Description
TransportContract	(S, PC), (PC, S)	TransportContractID, the *PrimeContractor* identifier and packages info are included.
DeliveryContract	(PC, SC_1), (SC_1, PC), (PC, SC_2), (SC_2, PC)	TransportContractID, DeliveryContractID, the *Subcontractor* identifier, packages info, are included.
DeliverDriver	(SC_1, U_{11}), (SC_2, U_{21})	DeliveryContractID, the UAV identifier and packages info are included.
PackageReceipt	(U_{11}, S), (U_{21}, U_{11})	This is used to prove that the packages has been relayed and it contains info related to the packages.

3.2 Issuing the contract

In this section we describe a procedures for issuing VC as a contract. A procedure between *Sender* and *PrimeContractor* is described below as an example.

As depicted in Figure 2, both *Sender* and *PrimeContractor* mutually issue TransportContractVC to each other. This process ensures that both players can prove that a contract has been issued between them. Additionally, a unique identifier, TransportContractID, is included in both contracts. Similar procedure for issuing VCs are carried between *PrimeContractor* and *Subcontractor*.

3.3 Relaying the packages

In this section we describe a procedures for relaying the packages using VCs. A procedure between *Sender* and UAV_{11} is described below as an example.

There are two features involved in relaying the packages. The first feature is mutual authentication before relaying packages. The second feature is to issue the receipt. In Figure 3, the flow diagram summarizing the two features is depicted.

Mutual Authentication For mutual authentication, both *Sender* and UAV_{11} create Verifiable Presentations (VPs) from their VCs. Both players create VPs to provide a minimum set of attributes for authentication. (1)UAV_{11} creates a VP from two VCs, namely the DeliveryContract VC issued by *PrimeContractor* and the DeliveryDriver VC issued by $Subcontractor_1$. (3)The *Sender* creates a VP from the TransportContract VC issued by *PrimeContractor*.

Both players confirm that they are collaborating under same *PrimeContractor*. *Sender* verifies a UAV by the chain of VCs in a VP. *Sender* confirms that the UAV is owned by *Subcontractor*, who has a contract with *PrimeContractor*, who has a contract with *Sender*. By using selective disclosure technique with zero-knowledge proofs and linking properties of Linked Data format, mutual authentication can be performed without showing other information, such as UAV identifiers.

Issuing a receipt *Sender* receives a receipt that can prove the package was accepted by UAV_{11}. The receipt is expressed as a PackageReceipt VC. (R.a) To ensure that *Sender* cannot determine which UAV is authenticating, UAV_{11} creates a fresh key pair for the PackageReceipt VC on each authentication. The new public key generated by UAV_{11} is conveyed to the *Sender* within a 'VP with Message'(1), which is a new data format we developed. The details will be described in the next paragraph. (R.b) UAV_{11} creates the PackageReceipt VC. (R.c) *Sender* verifies the PackageReceipt VC using the received public key to ensure it was created correctly.

We now describe our new data format for VP, called 'VP with Message', which is used in (1) in Figure 3. This data format allows Holder of VC to embed arbitrary messages in VP, which results in adding signature of knowledge to the message. Using this approach, we can embed new public key as the message in VP. This key will be used to verify the signature on PackageReceipt VC. This allows *Sender* to confirm that the receipt was created by the authenticated UAV.

3.4 Implementation

We created a demo application based on the proposed procedures. We adopt BBS+ signatures[3] to perform unlinkable authentication, and use Linked-Data based VC to execute zero-knowledge proofs necessary for selective disclosure[4]. For enhancing BBS+ signature, we extended from zkp-ld libraries [5] such as a VP with Message.

4 Conclusion

This study explores the applicability of VCs and VPs in a scenario where UAVs autonomously deliver packages. In the scenario, we have shown that VCs can

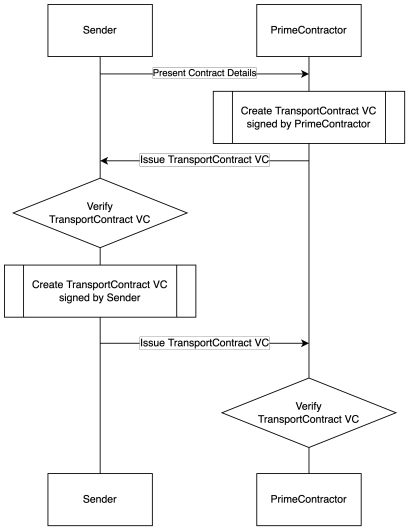

Fig. 2. Procedure of issuing a contract

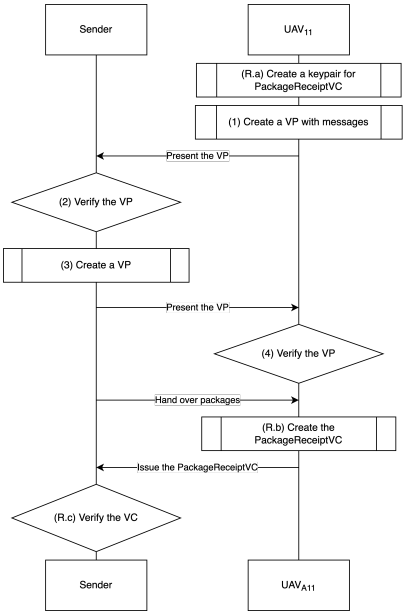

Fig. 3. Procedure of relaying packages

be utilized for package delivery contracts, authentication between UAVs, and issuing receipts between UAVs. We proposed a new data format for VP to enable issuing receipts by anonymous UAVs. Additionally, we have shown that the selective disclosure feature of VPs enables anonymous authentication. By implementing these features in a demo application, we have showed the feasibility and effectiveness of using VCs and VPs in UAV delivery systems. One challenge that arises is the inability to hide which *Subcontractor* the UAV belongs to. This is because the issuer of VCs cannot be hidden. This research highlights the potential for these technologies to enhance privacy in authentication.

References

1. Verifiable Credentials Data Model v1.1 W3C Recommendation 03 March 2022, https://www.w3.org/TR/vc-data-model/
2. European Union, Eu digital covid certificate, https://commission.europa.eu/live-work-travel-eu/coronavirus-response/safe-covid-19-vaccines-europeans/eu-digital-covid-certificate
3. Au, Susilo, and Mu, "Constant-size dynamic k-TAA," in SCN 06, ser. LNCS, R. D. Prisco and M. Yung, Eds., vol. 4116.
4. Dan Yamamoto, Yuji Suga, and Kazue Sako. Formalising linked-data based verifiable credentials for selective disclosure. In 2022 IEEE European Symposium on Security and Privacy, Workshops (EuroS&PW), pp. 52–65, 2022
5. Dan Yamamoto, zkp-ld (GitHub Organization), https://github.com/zkp-ld