Security analysis of a blockchain-based protocol for the certification of academic credentials

Marco Baldi, Franco Chiaraluce, Migelan Kodra, and Luca Spalazzi

Dipartimento di Ingegneria dell'Informazione
Università Politecnica delle Marche
Ancona, Italy, 60131
m.baldi@univpm.it, f.chiaraluce@univpm.it, migelankodra@yahoo.com, l.spalazzi@univpm.it

Abstract

We consider a blockchain-based protocol for the certification of academic credentials named Blockcerts, which is currently used worldwide for validating digital certificates of competence compliant with the Open Badges standard. We study the certification steps that are performed by the Blockcerts protocol to validate a certificate, and find that they are vulnerable to a certain type of impersonation attacks. More in detail, authentication of the issuing institution is performed by retrieving an unauthenticated issuer profile online, and comparing some data reported there with those included in the issued certificate. We show that, by fabricating a fake issuer profile and generating a suitably altered certificate, an attacker is able to impersonate a legitimate issuer and can produce certificates that cannot be distinguished from originals by the Blockcerts validation procedure. We also propose some possible countermeasures against an attack of this type, which require the use of a classic public key infrastructure or a decentralized identity system integrated with the Blockcerts protocol.

1 Introduction

The certification of competences and academic credentials plays a fundamental role in modern societies and everyday life. Classically, such a certification is performed through paper certificates containing seals and signatures. These documents, however, have no uniformity among different countries as well as no recognized digital equivalent and, most importantly, are subject to falsification. This is confirmed by several cases occurred in recent years, as for the dean of admissions at the Massachusetts Institute of Technology (MIT), who declared to have fabricated and lied about her own educational credentials for 28 years [1]. In this context, an internationally recognized standard for the digitalization and authentication of competence certificates (as transcripts, diplomas, etc.) and academic credentials becomes a must. Besides providing strong authentication mechanisms, such a system should also aim at establishing an internationally recognized format for portability and verification of competences.

The first step into verifying competences online and everywhere was made by the Mozilla Foundation [2] in collaboration with the McArthur Foundation [3], through the open-source project called *Open Badges* [4]. The scope of Open Badges is to provide organizations and institutions with a system for issuing digital badges to the competence owners in order to recognize not only their official learning but even transversal skills. For this purpose, certificates compliant with Open Badges are designed to provide a detailed profile of the recipient, including the so-called *Diploma Supplement* [5], collecting all the academic records of a student and giving this way a clear picture about the competences gained during the academic path. The Open

Badges standard hence provides a tool for implementing digital, enriched versions of competence certificates and academic credentials. However, the certification of the competences covered by one of these digital badges is out of the scope of the Open Badges standard itself.

Justified by its pervasiveness and huge potential, the blockchain technology has recently emerged as a tool to accelerate and facilitate the process of issuing and recognizing credentials in an increasingly digitised world [6]. Several proposals have already appeared in the literature, for example based on permissioned blockchains [7, 8], even with the goal of connecting learning records across different institutions [9]. In such a framework, the use of blockchain technology has also emerged as a valuable solution for the validation of Open Badges-compliant certificates [10]. Actually, the most widespread and internationally adopted blockchain-based system for the validation of these certificates was developed by the MIT Media Lab [11] in collaboration with Learning Machine [12], and is called *Blockcerts* [13]. In Italy there is an initiative, called Bestr [14], which aims at using the Blockcerts standard for the certification of diplomas issued by national universities.

According to the Blockcerts standard, Open Badges-compliant certificates are cryptographically signed by the issuer, while their certification data is written into a public blockchain, thus leveraging its immutability. Recipients can share them publicly on their social media profiles, personal websites, etc., and everyone can see the contents of the certificates with the possibility to verify their validity and authenticity through the blockchain. The use of the blockchain technology makes the process of verification globally accessible and instantaneous, with no need for long procedures and institutional bureaucratic correspondences. Another relevant point lays in the fact that Blockcerts passes from paper-based certificates to software-based certificates. A Blockcerts certificate is fully-machine readable since it is designed as software in JavaScript Object Notation (JSON) format [15]. Moreover, the fully decentralized architecture of the Blockcerts protocol makes it resilient to single point of failures, which may result from a temporary or permanent outage of the recipient or issuing institution digital services. Other solutions exist, like BCDiploma[1], which however do not rely on such a completely decentralized infrastructure. In fact, according to the BCDiploma approach, the blockchain is used to store the diplomas in encrypted form. Encryption is performed through a symmetric cipher using the combination of three keys: one for the recipient, one for the issuing institution and one for the service provider. Then, the retrieval and authentication of the diploma is based on the use of these three keys, thus resulting in a system that basically relies on a centralized infrastructure.

When analyzing and implementing blockchain-based systems, a very important issue concerns security [16], [17] that, indeed, is often not sufficiently explored. In this paper we review the steps of issuance and validation of a certificate compliant with the Blockcerts protocol, with the aim of analyzing its security and resistance to forgery. In particular, we focus on the certificate authentication process through the blockchain, which is designed to be decentralized and self-consistent. For this purpose, each certificate must contain all the necessary information for its validation through the blockchain, including a reference to the public key of the issuer to be used for its validation. We show that such a feature, although allowing decentralized validation, opens the door to possible forgery attacks. We first describe how an attack of this type can be mounted, and then we show its feasibility through a practical example.

For this purpose, we generate a forged though verifiable academic certificate. Such a forged certificate appears to be correctly issued by the *Università Politecnica delle Marche* according to the Blockcerts protocol, despite it has been fabricated without involving the aforementioned institution. In fact, by generating a valid blockchain key pair, and following a few simple steps, it was possible to impersonate the issuing institution and fabricate an apparently valid academic

[1] https://www.bcdiploma.com/

certificate. This is because the Blockcerts verification system does not check if the public keys are actually owned by the legitimate issuing institution or not. In order to prevent this type of attacks, we propose some countermeasures aimed at avoiding that a fake issuer profile can be accepted by the certificate verification protocol.

The paper is organized as follows. In Section 2 we describe the Blockcerts standard that is the object of our work. In Section 3 we describe a vulnerability in the Blockcerts certificate validation procedure and how it can be exploited to fabricate forged academic credentials. In Section 4 we describe some possible countermeasures aimed at preventing the aforementioned attack, while in Section 5 we provide some conclusive remarks.

2 Blockcerts

With the aim of having a global system for the verification of academic records, the development of Blockcerts [13] was initiated in 2015 as part of a research project by the MIT Media Lab [11] in collaboration with Learning Machine [12]. The Blockcerts project exploits the Open Badges framework [18, 19] jointly with the blockchain technology in order to realize a global, decentralized notary. Consequently, Blockcerts adds several features to the Open Badges specification, namely [20]: i) Tamper evidence, ii) Issuer and recipient ownership, iii) Flexible form factor (which means that a flexible document display is embedded in the Blockcert JSON file), iv) Online and offline sharing with verification and v) Independent verification. The project was officially launched in 2016 and all the reference libraries were published under the MIT Open Source Licence, making the code accessible and free of charge. Therefore, Blockcerts is defined as an open standard for creating, issuing, viewing and verifying blockchain-based certificates.

Basically, the issuer can autonomously create a structured Blockcerts certificate, sign it and certify its integrity by storing a hash digest of the certificate within a blockchain transaction. Then, the issuer can send the recipient a copy of the signed Blockerts certificate that can be shared in social networks, via e-mail, etc. Anyone accessing the certificate can verify its integrity using an open platform called *Blockcerts Universal Verifier* [13], that performs a blockchain-based integrity check.

The whole lifecycle of a Blockcerts-based certificate is schematically described in Figure 1. After successfully completing her/his studies, the student is asked to provide a public blockchain address by the issuing institution. For this purpose, the student generates a private/public keypair for the used blockchain, and then computes her/his public address as the output of a one-way function applied to her/his public key. At the same time (or before) a Blockcerts-compliant certificate template is created by the issuing institution. Then, a new certificate is issued and released to the student, along with a blockchain transaction to the student's public address that enables verification of the issued certificate.

2.1 Blockcerts Certificate Design

Each Blockcerts certificate contains structured data concerning the certificate itself, its issuer and its recipient (see Figure 2.a).

Information about the issuer is contained in the Issuer Profile, according to the Open Badge standard. Among the various fields, for the purpose of this work, an important one is the issuer id, which is the URL of a web page containing the issuer information in JSON format. As we can see from the example reported in Figure 3, such information includes the institution name, homepage, logo, e-mail address, etc., and, importantly, the public keys claimed by the issuer. Each key has a timestamp corresponding to its creation and possibly the timestamp

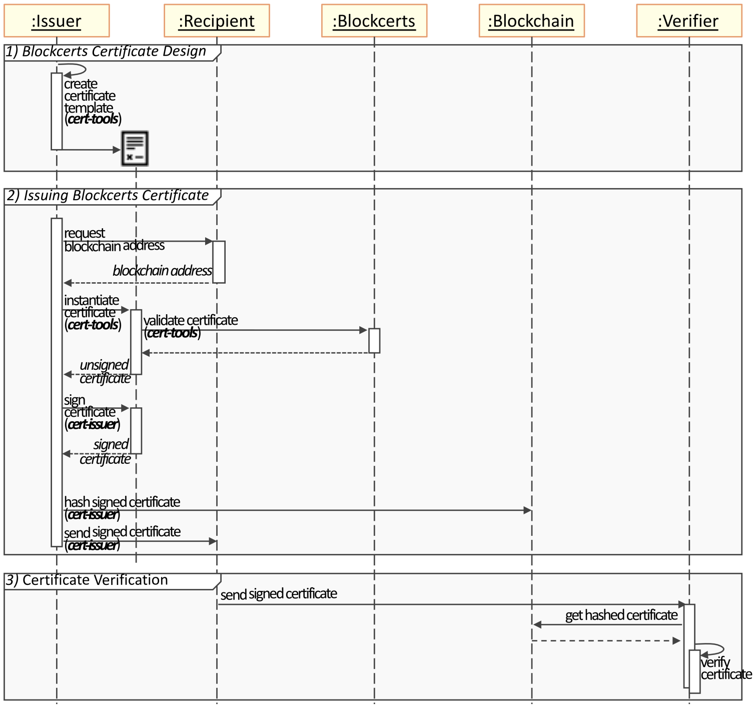
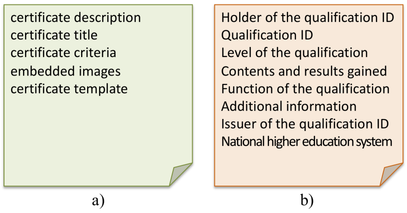
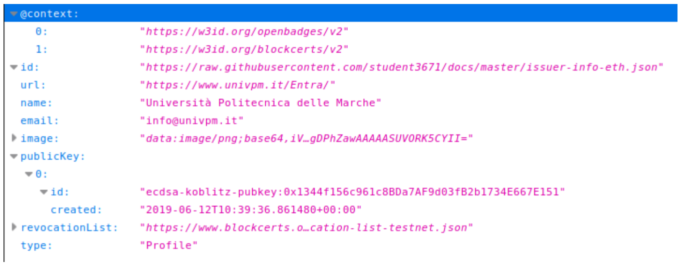

:Issuer
:Recipient
:Blockcerts
:Blockchain
:Verifier
1) Blockcerts Certificate Design
create certificate template (cert-tools)
2) Issuing Blockcerts Certificate
request blockchain address
blockchain address
instantiate certificate (cert-tools)
validate certificate (cert-tools)
unsigned certificate
sign certificate (cert-issuer)
signed certificate
hash signed certificate (cert-issuer)
send signed certificate (cert-issuer)
3) Certificate Verification
send signed certificate
get hashed certificate
verify certificate

Figure 1: Sequence flow of the Blockcerts system.



certificate description
certificate title
certificate criteria
embedded images
certificate template

Holder of the qualification ID
Qualification ID
Level of the qualification
Contents and results gained
Function of the qualification
Additional information
Issuer of the qualification ID
National higher education system



```
@context:
   0:              "https://w3id.org/openbadges/v2"
   1:              "https://w3id.org/blockcerts/v2"
id:                "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
url:               "https://www.univpm.it/Entra/"
name:              "Università Politecnica delle Marche"
email:             "info@univpm.it"
image:             "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
publicKey:
   0:
      id:          "ecdsa-koblitz-pubkey:0x1344f156c961c8BDa7AF9d03fB2b1734E667E151"
      created:     "2019-06-12T10:39:36.861480+00:00"
revocationList:    "https://www.blockcerts.o…cation-list-testnet.json"
type:              "Profile"
```

Figure 2: a) Content of a certificate. b) Diploma supplement.

Figure 3: Hosted issuer profile.

corresponding to its expiration. Regarding information about the recipient, it concerns: name, email, and blockchain public key.

As mentioned in the Introduction, the Blockcerts standard also allows embedding the so-called *diploma supplement* into a certificate. The diploma supplement is produced by higher education institutions according to international and European recommendations, and contains eight sections (see Figure 2.b). It includes additional information about the course attended (e.g., results for each exam/course with date and outcome) and the competences achieved by the recipient of the certificate. The diploma supplement is recognized as an official document accompanying a higher education diploma, providing a standardized description of the nature, level, context, content and status of the studies completed by its holder.

2.2 Issuing Blockcerts Certificate

All the preliminary steps required for issuing certificates compliant with the Blockcerts standard can be implemented through the *cert-tools* open source software [21], which allows generating a certificate template first, and then instantiating such a template into one or more certificates. This, after validation, allows generating an unsigned certificate, which then has to be signed and written into the blockchain network, besides sending a copy to the recipient. These steps can be performed through the *cert-issuer* software tool [22]. Delivery of the certificate is performed by creating a transaction from the issuing institution to the certificate recipient on the Bitcoin or Ethereum blockchain that includes the hash of the certificate itself. While it is possible to issue one certificate with one Bitcoin/Ethereum transaction, a solution for reducing the amount of data written to the blockchain is that of using one Bitcoin/Ethereum transaction to issue a batch of certificates. This is possible through the generation of a Merkle tree of certificate hashes, in such a way that only the Merkle tree root has to be written into the blockchain.

2.3 Certificate Authenticity Verification

Through the verification process, anyone can check the authenticity of a certificate, having clear information about the institution that issued it and a proof that the certificate was actually issued to the claiming recipient. The certificate verification process starts with the following three steps:

1. Verification that the hash digest of the certificate matches the value in the receipt.

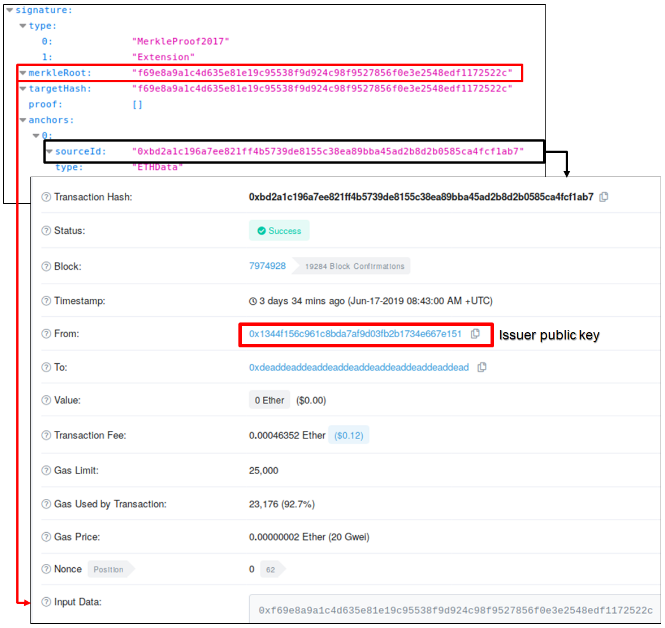
2. Verification that the Merkle path is valid.

3. Verification that the Merkle root stored into the blockchain matches the value in the receipt.

Through the above steps, anyone can check that a certificate has not been tampered since its issuing. The next important step of the verification process is authenticating the certificate issuer, i.e., verifying the identity of the issuing institution. This is achieved by verifying that the signing key for the blockchain transaction through which the certificate was issued corresponds to the issuer public key, and that it was valid when the transaction took place. This uses the timestamp and input address from the blockchain transaction details, and the issuer information provided along with the *Issuer Profile*, described in Section 2.1. For this purpose, the blockchain transaction id is extracted from the certificate receipt. The transaction id allows verifying that the transaction was actually registered into the blockchain and retrieving the corresponding transaction details. The issuer public key can be found within such details, as shown in Figure 4.

Then, the *issuer_id* is extracted from the certificate to retrieve the *Hosted Issuer Profile* from the corresponding url. The issuer public key included in the *Hosted Issuer Profile* is then compared with the one found in the blockchain transaction, as shown in Figure 5.

If the two keys do not coincide, an error is returned and the certificate is considered invalid. Otherwise, the timestamps are checked to prove that the key was valid at the time of the transaction. In addition, if the public key included in the issuer profile has an expiration date, it is checked that the transaction did not take place after that date. If all these verification steps succeed, then the certificate is considered as valid. All the steps of the issuer identity verification are schematically described in Figure 6.



```
signature:
  type:
    0:          "MerkleProof2017"
    1:          "Extension"
  merkleRoot:   "f69e8a9a1c4d635e81e19c95538f9d924c98f9527856f0e3e2548edf1172522c"
  targetHash:   "f69e8a9a1c4d635e81e19c95538f9d924c98f9527856f0e3e2548edf1172522c"
  proof:        []
  anchors:
    0:
      sourceId: "0xbd2a1c196a7ee821ff4b5739de8155c38ea89bba45ad2b8d2b0585ca4fcf1ab7"
      type:     "ETHData"
```

Transaction Hash:	0xbd2a1c196a7ee821ff4b5739de8155c38ea89bba45ad2b8d2b0585ca4fcf1ab7
Status:	Success
Block:	7974928 19284 Block Confirmations
Timestamp:	3 days 34 mins ago (Jun-17-2019 08:43:00 AM +UTC)
From:	0x1344f156c961c8bda7af9d03fb2b1734e667e151
To:	0xdeaddeaddeaddeaddeaddeaddeaddeaddeaddead
Value:	0 Ether ($0.00)
Transaction Fee:	0.00046352 Ether ($0.12)
Gas Limit:	25,000
Gas Used by Transaction:	23,176 (92.7%)
Gas Price:	0.00000002 Ether (20 Gwei)
Nonce Position	0 62
Input Data:	0xf69e8a9a1c4d635e81e19c95538f9d924c98f9527856f0e3e2548edf1172522c

Figure 4: Extraction of the issuer public key from the transaction details.

3 Fabricating academic credentials

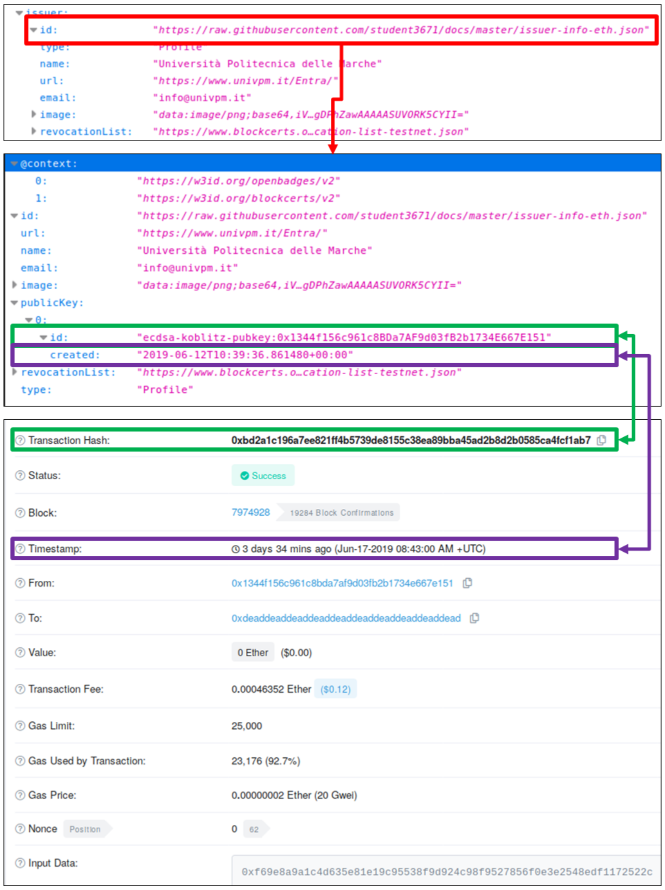
In this section we show that apparently valid certificates issued by any institution can indeed be fabricated by a malicious attacker. This is basically due to an intrinsic vulnerability of the verification process described in Section 2.3. In fact, the Blockcerts protocol does not verify that the *issuer_id* extracted from a certificate actually points to a web domain that is owned by the legitimate issuing institution. This allows hijacking of the verifier towards a fake issuer profile, which can perfectly resemble the one of the legitimate institution.

In our experiment, a fake issuer profile was created for the Università Politecnica delle Marche and hosted on a Github domain. The public key included in such a profile has no relation with the real institution, since it was auto-generated for the purposes of this work. During the verification process, the Blockcerts protocol checks the public key on the blockchain transaction corresponding to the certificate, and compares it with the key included in the issuer profile published online. As we show next, this brings to a successful verification through Blockcerts, and to a forged certificate that is practically indistinguishable from a legitimate one. This results in the fabrication of a certificate for the Italian Laurea Magistrale in Electronic Engineering by impersonating the issuing institution Università Politecnica delle Marche.

3.1 Creation of the certificate

Two Ethereum blockchain keypairs were first generated: one for the issuer (university) and one for the recipient (student). After that, a Blockcerts-compliant certificate has been designed. This step can obviously be skipped if the issuer already has defined its own Blockcerts-compliant certificates.

A new certificate is then issued to the student, including the blockchain public key of the student, the student's personal information and a diploma supplement. The correctness of



```
issuer:
  id:              "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
  type:            "Profile"
  name:            "Università Politecnica delle Marche"
  url:             "https://www.univpm.it/Entra/"
  email:           "info@univpm.it"
  image:           "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
  revocationList:  "https://www.blockcerts.o…cation-list-testnet.json"
```

```
@context:
  0:             "https://w3id.org/openbadges/v2"
  1:             "https://w3id.org/blockcerts/v2"
id:              "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
url:             "https://www.univpm.it/Entra/"
name:            "Università Politecnica delle Marche"
email:           "info@univpm.it"
image:           "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
publicKey:
  0:
    id:          "ecdsa-koblitz-pubkey:0x1344f156c961c8BDa7AF9d03fB2b1734E667E151"
    created:     "2019-06-12T10:39:36.861480+00:00"
revocationList:  "https://www.blockcerts.o…cation-list-testnet.json"
type:            "Profile"
```

```
Transaction Hash:             0xbd2a1c196a7ee821ff4b5739de8155c38ea89bba45ad2b8d2b0585ca4fcf1ab7
Status:                       Success
Block:                        7974928  19284 Block Confirmations
Timestamp:                    3 days 34 mins ago (Jun-17-2019 08:43:00 AM +UTC)
From:                         0x1344f156c961c8bda7af9d03fb2b1734e667e151
To:                           0xdeaddeaddeaddeaddeaddeaddeaddeaddeaddead
Value:                        0 Ether  ($0.00)
Transaction Fee:              0.00046352 Ether  ($0.12)
Gas Limit:                    25,000
Gas Used by Transaction:      23,176 (92.7%)
Gas Price:                    0.00000002 Ether (20 Gwei)
Nonce  Position               0  62
Input Data:                   0xf69e8a9a1c4d635e81e19c95538f9d924c98f9527856f0e3e2548edf1172522c
```

Figure 5: Issuer identity verification (getting hosted issuer id, comparing public keys and checking timestamps consistency).

this information is validated according to the Blockcerts and Open Badges standards, after which an unsigned certificate is obtained. Such a certificate is then signed with the issuer keypair and issued through a transaction on the Ethereum blockchain. The main content of such a blockchain transaction is the certificate Merkle root. In our case, since a single certificate was issued, the Merkle root coincides with the certificate hash digest. This way, a signed Blockcerts-compliant certificate is obtained, as shown in Figure 7. Note that the signed certificate includes the blockchain receipt, providing all the necessary details for retrieving the blockchain transaction and performing the blockchain-based verification of the certificate according to the Blockcerts standard.

This signed certificate can be verified through any web-based or stand-alone tool compliant with the Blockcerts protocol. Let us use for this purpose the Blockcerts Universal Verifier[2]. The outcome of the verification, so performed, is shown in Figure 8. The verification process indeed ended with a positive result. Moreover, the verification window reports the issuing university logo and the title of the certificate (in this case: *Master's Degree in Electronic Engineering*), followed by the name of the recipient, the issuance date and the name of the

[2] https://www.blockcerts.org/

```
boolean verifyCertificate(Certificate c)
{
        BlockchainTransactionID transID = getBlockchainTransactionID(c);

        // get info about issuer from URL link contained into the certificate
        IssuerID                issuerID = getIssuerID(c);
        URL                     profileURL = getHostedIssuerProfileURL(issuerID);
        // read the profile following the URL link
        Profile                 profile = httpGET(profileURL);
        PublicKey               pkey = getPublicKey(profile);
        BlockchainAddress       add1 = computeBlockchainAddress(pkey);
        Timestamp               t1 = getTimestamp(pkey);

        // get info about issuer from the blockchain transaction
        // read the transaction from blockchain
        BlockchainTransaction trans = blockchainGET(transID);
        BlockchainAddress     add2 = getBlockchainAddress(trans);
        Timestamp   t2 = getTimestamp(trans);

        if (add1==add2) && (t1 <= t2)
                return true // the issuer is valid
        else
                return false // the issuer is not valid
}
```

Figure 6: Blockcerts issuer verification process.

issuing institution. The diploma supplement and its contents can be visualized as well. The details of the corresponding Ethereum transaction[3] are reported in Figure 9.

3.2 Analysis of the verification process

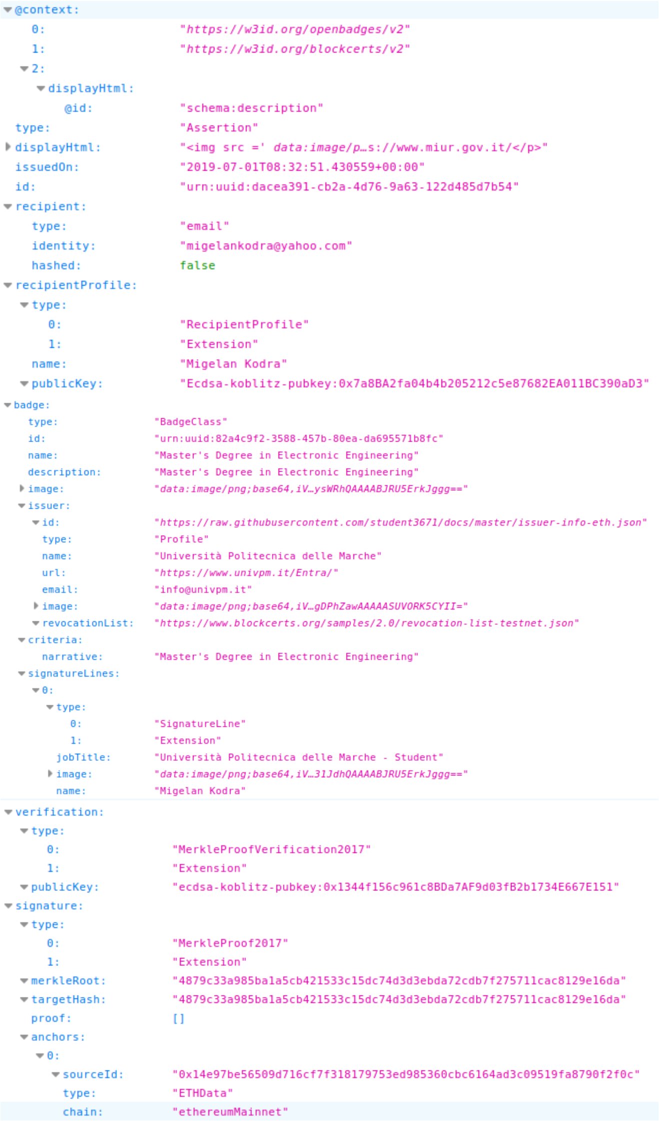
Let us describe how it is possible that a fabricated certificate passes all the verification steps required by the Blockcerts protocol by analyzing them in detail.

As shown in Figure 8, the first verification step consists in reading the transaction id, through which the details reported in Figure 9 are retrieved. Then, a local hash digest of the certificate is computed, and the remote hash digest is retrieved from the associated blockchain transaction.

The next verification step consists in getting the issuer profile referenced by the issuer_id. This is a crucial point for the successful verification of a fabricated certificate, and is schematically described in Figure 10. As we notice from the figure, the issuer_id points to a custom url that has no relation with the legitimate issuer. This is the point where hijacking of the verification occurs, and the use of a fabricated issuer profile is enforced. The next step consists in parsing the issuer keys, that is, extracting the issuer public key from the public issuer profile, which in our case was illegally fabricated.

The verification process continues with a second block of steps concerning the hash digest comparison. The first of these steps verifies that the hash of the certificate locally computed coincides with that included in the certificate receipt. This proves that the certificate has not been modified since its issuance. In the next step, the system compares the Merkle root value written on the certificate receipt with the Merkle root written on the blockchain. As a next step, the certificate receipt is checked to verify that the certificate under analysis is part of the Merkle tree. As mentioned above, in our case, only one certificate was issued and its hash digest corresponds with the Merkle root value. This is denoted by leaving the *proof* field empty in the certificate receipt. In such a case, the system realizes that only one certificate was issued

[3] Available at https://etherscan.io/tx/0x14e97be56509d716cf7f318 179753ed985360cbc6164ad3c09519fa8790f2f0c



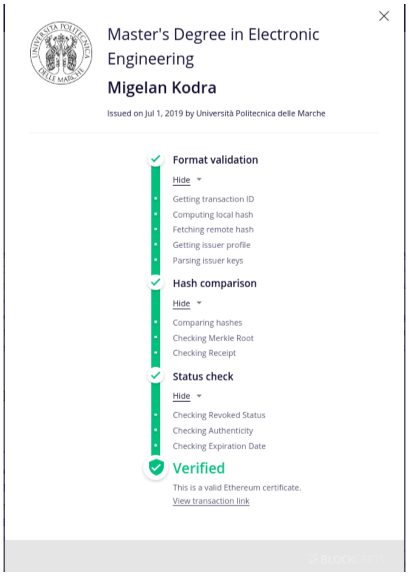
```
@context:
  0:                    "https://w3id.org/openbadges/v2"
  1:                    "https://w3id.org/blockcerts/v2"
  2:
    displayHtml:
      @id:              "schema:description"
type:                   "Assertion"
displayHtml:            "<img src =' data:image/p…s://www.miur.gov.it/</p>"
issuedOn:               "2019-07-01T08:32:51.430559+00:00"
id:                     "urn:uuid:dacea391-cb2a-4d76-9a63-122d485d7b54"
recipient:
  type:                 "email"
  identity:             "migelankodra@yahoo.com"
  hashed:               false
recipientProfile:
  type:
    0:                  "RecipientProfile"
    1:                  "Extension"
  name:                 "Migelan Kodra"
  publicKey:            "Ecdsa-koblitz-pubkey:0x7a8BA2fa04b4b205212c5e87682EA011BC390aD3"
badge:
  type:                 "BadgeClass"
  id:                   "urn:uuid:82a4c9f2-3588-457b-80ea-da695571b8fc"
  name:                 "Master's Degree in Electronic Engineering"
  description:          "Master's Degree in Electronic Engineering"
  image:                "data:image/png;base64,iV…ysWRhQAAAABJRU5ErkJggg=="
  issuer:
    id:                 "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
    type:               "Profile"
    name:               "Università Politecnica delle Marche"
    url:                "https://www.univpm.it/Entra/"
    email:              "info@univpm.it"
    image:              "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
    revocationList:     "https://www.blockcerts.org/samples/2.0/revocation-list-testnet.json"
  criteria:
    narrative:          "Master's Degree in Electronic Engineering"
  signatureLines:
    0:
      type:
        0:              "SignatureLine"
        1:              "Extension"
      jobTitle:         "Università Politecnica delle Marche - Student"
      image:            "data:image/png;base64,iV…31JdhQAAAABJRU5ErkJggg=="
      name:             "Migelan Kodra"
verification:
  type:
    0:                  "MerkleProofVerification2017"
    1:                  "Extension"
  publicKey:            "ecdsa-koblitz-pubkey:0x1344f156c961c8BDa7AF9d03fB2b1734E667E151"
signature:
  type:
    0:                  "MerkleProof2017"
    1:                  "Extension"
  merkleRoot:           "4879c33a985ba1a5cb421533c15dc74d3d3ebda72cdb7f275711cac8129e16da"
  targetHash:           "4879c33a985ba1a5cb421533c15dc74d3d3ebda72cdb7f275711cac8129e16da"
  proof:                []
  anchors:
    0:
      sourceId:         "0x14e97be56509d716cf7f318179753ed985360cbc6164ad3c09519fa8790f2f0c"
      type:             "ETHData"
      chain:            "ethereumMainnet"
```

Figure 7: Blockcerts signed certificate.

and verifies that the Merkle root values in the certificate and the blockchain are correct and coincide with the certificate hash digest.

Instead, when a batch of certificates is issued, the *proof* field is filled with the necessary values to create the path from the current certificate to the Merkle root. An illustrative example is shown in Figure 11, in which the Merkle path is colored in orange. With this information, the system is able to calculate the final value of the Merkle root of the whole batch of certificates. If the certificate is part of the batch, then the calculated Merkle root value is the same as the value written on the certificate. Otherwise, an error occurs, meaning that the certificate is not part of the batch, and therefore, is not valid.

At this point, it has been proved that the certificate has not been modified since its issuance



Master's Degree in Electronic Engineering

Migelan Kodra

Issued on Jul 1, 2019 by Università Politecnica delle Marche

✓ **Format validation**

Hide

- Getting transaction ID
- Computing local hash
- Fetching remote hash
- Getting issuer profile
- Parsing issuer keys

✓ **Hash comparison**

Hide

- Comparing hashes
- Checking Merkle Root
- Checking Receipt

✓ **Status check**

Hide

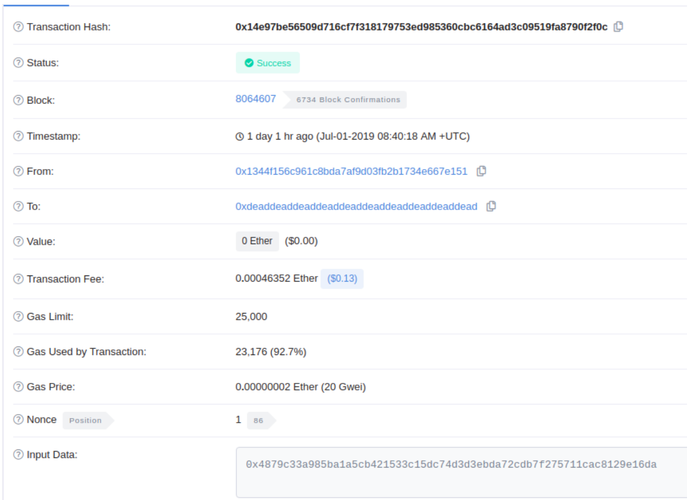
- Checking Revoked Status
- Checking Authenticity
- Checking Expiration Date

✓ **Verified**

This is a valid Ethereum certificate.

View transaction link

Figure 8: Certificate verification through the Blockcerts Universal Verifier.



Transaction Hash:	0x14e97be56509d716cf7f318179753ed985360cbc6164ad3c09519fa8790f2f0c
Status:	Success
Block:	8064607 6734 Block Confirmations
Timestamp:	1 day 1 hr ago (Jul-01-2019 08:40:18 AM +UTC)
From:	0x1344f156c961c8bda7af9d03fb2b1734e667e151
To:	0xdeaddeaddeaddeaddeaddeaddeaddeaddeaddead
Value:	0 Ether ($0.00)
Transaction Fee:	0.00046352 Ether ($0.13)
Gas Limit:	25,000
Gas Used by Transaction:	23,176 (92.7%)
Gas Price:	0.00000002 Ether (20 Gwei)
Nonce Position	1 86
Input Data:	0x4879c33a985ba1a5cb421533c15dc74d3d3ebda72cdb7f275711cac8129e16da

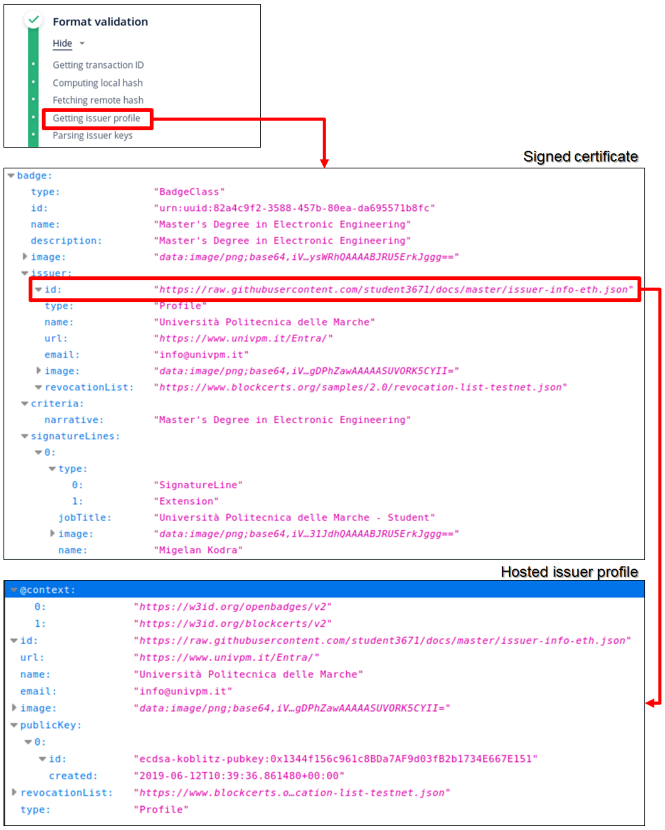
Figure 9: Ethereum blockchain transaction details.

and that it is actually written on the blockchain. The next block of verification steps, named *status check*, concerns the certificate authentication. The first one of these checks concerns the revocation status, and is aimed at verifying that the certificate has not been revoked by the issuer. In fact, a list of revoked certificates is available online along with the issuer profile, as shown in Figure 12. The system goes through such a list to check if it includes the certificate under analysis or not. In case the certificate identifier is found in the list of revoked certificates, an error message is returned and the verification fails, otherwise the system continues with the next verification step.

The subsequent step, named authenticity checking step, is a very crucial element in the verification process. This step aims at verifying that the certificate was actually issued by the claimed institution. According to the Blockcerts standard, the certificate authenticity is checked as explained in Section 2.3. For the certificate under analysis, the steps performed for checking its authenticity are schematically described in Figure 13.

For this purpose, the hosted_id field is first read from the certificate, and the corresponding public key is retrieved from the issuer profile available online, at the web address specified in the hosted_id field. Such a public key is then compared with the one reported in the blockchain transaction. If the two keys coincide, the system checks the timestamps as explained before. If such a check is successful, verification proceeds by checking if the certificate has an expiry date, after which the verification process is completed.

In our case, the fabricated certificate was able to pass all the verification steps, and is therefore considered as a valid Blockcerts-compliant certificate. This proves that such a protocol does not allow distinguishing the legitimate issuer from someone impersonating it. This is due to the fact that the Blockcerts standard does not require any verification that the keys used to sign a certificate are actually owned by the legitimate issuing institution. For this reason, everyone creating a new keypair can sign a Blockcerts-compliant certificate and impersonate



```
badge:
  type:                 "BadgeClass"
  id:                   "urn:uuid:82a4c9f2-3588-457b-80ea-da695571b8fc"
  name:                 "Master's Degree in Electronic Engineering"
  description:          "Master's Degree in Electronic Engineering"
  image:                "data:image/png;base64,iV…ysWRhQAAAABJRU5ErkJggg=="
  issuer:
    id:                 "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
    type:               "Profile"
    name:               "Università Politecnica delle Marche"
    url:                "https://www.univpm.it/Entra/"
    email:              "info@univpm.it"
    image:              "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
    revocationList:     "https://www.blockcerts.org/samples/2.0/revocation-list-testnet.json"
  criteria:
    narrative:          "Master's Degree in Electronic Engineering"
  signatureLines:
    0:
      type:
        0:              "SignatureLine"
        1:              "Extension"
      jobTitle:         "Università Politecnica delle Marche - Student"
      image:            "data:image/png;base64,iV…31JdhQAAAABJRU5ErkJggg=="
      name:             "Migelan Kodra"
```

```
@context:
  0:                "https://w3id.org/openbadges/v2"
  1:                "https://w3id.org/blockcerts/v2"
id:                 "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
url:                "https://www.univpm.it/Entra/"
name:               "Università Politecnica delle Marche"
email:              "info@univpm.it"
image:              "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
publicKey:
  0:
    id:             "ecdsa-koblitz-pubkey:0x1344f156c961c8BDa7AF9d03fB2b1734E667E151"
      created:      "2019-06-12T10:39:36.861480+00:00"
revocationList:     "https://www.blockcerts.o…cation-list-testnet.json"
type:               "Profile"
```

Figure 10: Getting the issuer profile.

the legitimate issuing institution.

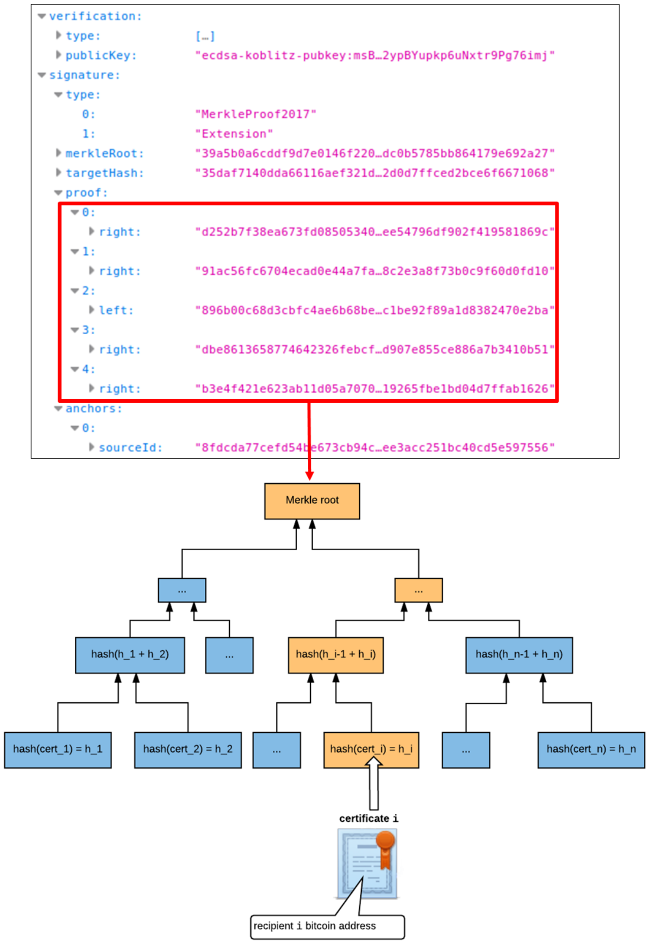
4 Possible countermeasures

The vulnerability of the Blockcerts standard described in the previous sections builds upon the lack of any certification about the ownership of the keys used for signing issued certificates. In order to prevent forgery attacks exploiting such a vulnerability, suitable countermeasures must be adopted by introducing a mechanism to check that such keys indeed correspond to the digital identity of the legitimate issuing institution.

A natural solution of this type could be replacing the issuer profile referenced from the *issuer_id* field with a digital certificate containing the public key of the issuing institution, and released by an accredited certification authority. In this way, the authenticity of the certificate can be checked through a classic Public Key Infrastructure (PKI), and the online information about the issuer retrieved through the *issuer_id* is certified.

Starting from July 2016, the electronic identification, authentication and trust services (eIDAS) came into effect in the European Union. The eIDAS regulation defines three types of electronic signatures [23]:

1. Electronic Signature: data in electronic form which is attached to or logically associated



```
▾ verification:
  ▸ type:               […]
  ▸ publicKey:          "ecdsa-koblitz-pubkey:msB…2ypBYupkp6uNxtr9Pg76imj"
▾ signature:
  ▾ type:
      0:                "MerkleProof2017"
      1:                "Extension"
  ▸ merkleRoot:         "39a5b0a6cddf9d7e0146f220…dc0b5785bb864179e692a27"
  ▸ targetHash:         "35daf7140dda66116aef321d…2d0d7ffced2bce6f6671068"
  ▾ proof:
    ▾ 0:
      ▸ right:          "d252b7f38ea673fd08505340…ee54796df902f419581869c"
    ▾ 1:
      ▸ right:          "91ac56fc6704ecad0e44a7fa…8c2e3a8f73b0c9f60d0fd10"
    ▾ 2:
      ▸ left:           "896b00c68d3cbfc4ae6b68be…c1be92f89a1d8382470e2ba"
    ▾ 3:
      ▸ right:          "dbe8613658774642326febcf…d907e855ce886a7b3410b51"
    ▾ 4:
      ▸ right:          "b3e4f421e623ab11d05a7070…19265fbe1bd04d7ffab1626"
  ▾ anchors:
    ▾ 0:
      ▸ sourceId:       "8fdcda77cefd54be673cb94c…ee3acc251bc40cd5e597556"
```

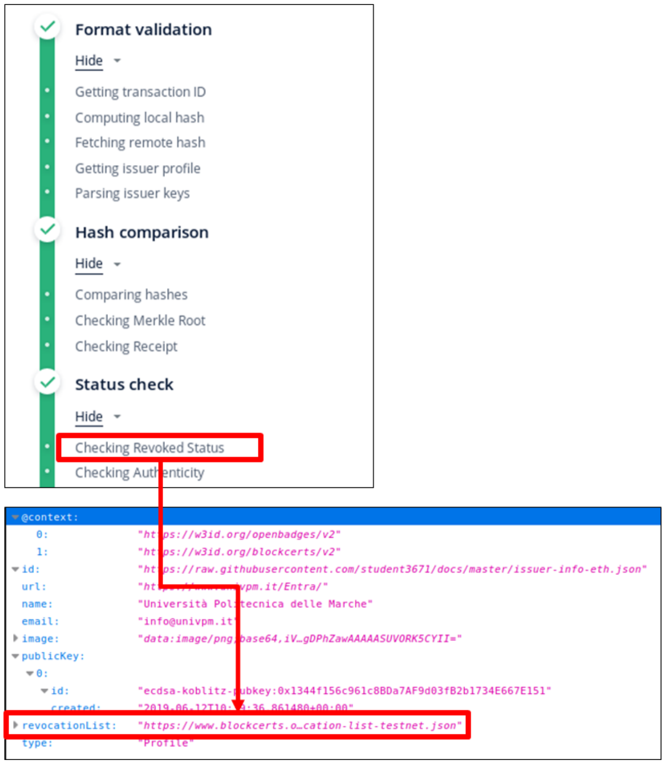
Figure 11: Merkle tree path for the verification of a certificate.

with other data in electronic form and which is used by the signatory to sign.

2. Advanced Electronic Signature: an electronic signature which meets the following requirements: a) it is uniquely linked to the signatory, b) it is capable of identifying the signatory, c) it is produced using electronic signature creation data that the signatory can use, with a high level of confidence, under his sole control, and d) it is linked to the data signed therewith in such a way that any subsequent change in the data is detectable.

3. Qualified Electronic Signature: an advanced electronic seal, which is created by a qualified electronic seal creation device, and that is based on a qualified certificate for electronic seal.

The only type of signatures universally recognized by all EU states and given the equivalent legal effect of a handwritten signature are qualified electronic signatures. The eIDAS regulation also provides the requirements for qualified certificates. Since most of the blockchain platforms (like the Ethereum blockchain used in our case) use Elliptic Curve Digital Signature Algorithm (ECDSA) signatures, that is based on Elliptic Curves Cryptography (ECC), a natural solution



Format validation

Hide

- Getting transaction ID
- Computing local hash
- Fetching remote hash
- Getting issuer profile
- Parsing issuer keys

Hash comparison

Hide

- Comparing hashes
- Checking Merkle Root
- Checking Receipt

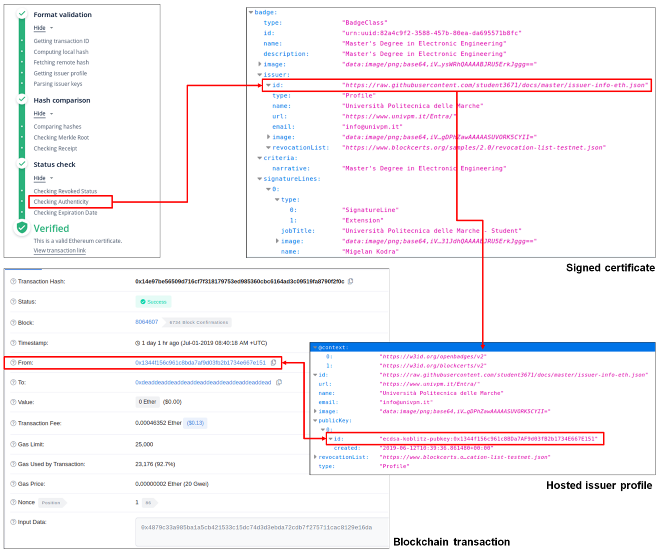
Status check

Hide

- Checking Revoked Status
- Checking Authenticity

```
@context:
    0:                "https://w3id.org/openbadges/v2"
    1:                "https://w3id.org/blockcerts/v2"
id:                   "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
url:                  "https://www.univpm.it/Entra/"
name:                 "Università Politecnica delle Marche"
email:                "info@univpm.it"
image:                "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
publicKey:
  0:
    id:               "ecdsa-koblitz-pubkey:0x1344f156c961c8BDa7AF9d03fB2b1734E667E151"
    created:          "2019-06-12T10:39:36.861480+00:00"
revocationList:       "https://www.blockcerts.o…cation-list-testnet.json"
type:                 "Profile"
```

Figure 12: Checking revocation of a certificate.



Format validation

Hide

- Getting transaction ID
- Computing local hash
- Fetching remote hash
- Getting issuer profile
- Parsing issuer keys

Hash comparison

Hide

- Comparing hashes
- Checking Merkle Root
- Checking Receipt

Status check

Hide

- Checking Revoked Status
- Checking Authenticity
- Checking Expiration Date

Verified

This is a valid Ethereum certificate.

View transaction link

```
badge:
  type:               "BadgeClass"
  id:                 "urn:uuid:82a4c9f2-3588-457b-80ea-da695571b8fc"
  name:               "Master's Degree in Electronic Engineering"
  description:        "Master's Degree in Electronic Engineering"
  image:              "data:image/png;base64,iV…ysWRhQAAAABJRU5ErkJggg=="
  issuer:
    id:               "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
    type:             "Profile"
    name:             "Università Politecnica delle Marche"
    url:              "https://www.univpm.it/Entra/"
    email:            "info@univpm.it"
    image:            "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
    revocationList:   "https://www.blockcerts.org/samples/2.0/revocation-list-testnet.json"
  criteria:
    narrative:        "Master's Degree in Electronic Engineering"
  signatureLines:
    0:
      type:
        0:            "SignatureLine"
        1:            "Extension"
      jobTitle:       "Università Politecnica delle Marche - Student"
      image:          "data:image/png;base64,iV…31JdhQAAAABJRU5ErkJggg=="
      name:           "Migelan Kodra"
```

Transaction Hash:	0x14e97be56509d716cf7f318179753ed985360cbc6164ad3c09519fa8790f2f0c
Status:	Success
Block:	8064607 6734 Block Confirmations
Timestamp:	1 day 1 hr ago (Jul-01-2019 08:40:18 AM +UTC)
From:	0x1344f156c961c8bda7af9d03fb2b1734e667e151
To:	0xdeaddeaddeaddeaddeaddeaddeaddeaddeaddead
Value:	0 Ether ($0.00)
Transaction Fee:	0.00046352 Ether ($0.13)
Gas Limit:	25,000
Gas Used by Transaction:	23,176 (92.7%)
Gas Price:	0.00000002 Ether (20 Gwei)
Nonce Position	1 86
Input Data:	0x4879c33a985ba1a5cb421533c15dc74d3d3ebda72cdb7f275711cac8129e16da

```
@context:
    0:                "https://w3id.org/openbadges/v2"
    1:                "https://w3id.org/blockcerts/v2"
id:                   "https://raw.githubusercontent.com/student3671/docs/master/issuer-info-eth.json"
url:                  "https://www.univpm.it/Entra/"
name:                 "Università Politecnica delle Marche"
email:                "info@univpm.it"
image:                "data:image/png;base64,iV…gDPhZawAAAAASUVORK5CYII="
publicKey:
  0:
    id:               "ecdsa-koblitz-pubkey:0x1344f156c961c8BDa7AF9d03fB2b1734E667E151"
    created:          "2019-06-12T10:39:36.861480+00:00"
revocationList:       "https://www.blockcerts.o…cation-list-testnet.json"
type:                 "Profile"
```

Figure 13: Checking the authenticity of a certificate.

would be using an ECC-based X.509 certificate format to link the issuer identity with its ECC-based public key.

This solution provides an effective way to overcome the Blockcerts issuer identity verification problem and to give a legal value to the Blockcerts academic certificate, owing to eIDAS compliance. On the other hand, the ECC-based certificate is signed by a Trusted Service Provider (TSP), and validity of the certificate relies on validity of the TSP signature. The main drawback of such a solution is in the fact that, this way, the verification system is no longer fully decentralized, needing a Certificate Authority (CA) to verify the identity of the issuer and sign the ECC-based certificate.

In order to overcome such a drawback, Decentralized Identifiers (DIDs) could be considered as an alternative solution. In fact, there are currently working groups [24] in the World Wide Web Consortium (the *W3C Community Group on Decentralized Identifier* and the *W3C Working Group on Verifiable Claims*) and in the Decentralized Identity Foundation (the DIF Working Group on DID Auth.) with the goal to achieve a common understanding of the general architecture of decentralized identity systems based on Blockchain and Distributed Ledger Technology (DLT), and to develop standards that enable interoperability between different implementations even on different DLT platforms, while following privacy and security-by-design principles. Although these initiatives still do not provide consolidated standards and practices, in the mid term they are expected to represent an effective solution to restore the completely decentralized nature of Blockcerts and similar protocols while countering forgery attacks like those described in this paper.

5 Conclusion

We have analyzed a blockchain-based protocol for the certification of academic credentials named Blockcerts, which aims at certifying digital certificates compliant with the Open Badges standard through a public blockchain. We have reviewed all the steps that are required for the creation of Open Badges-compliant digital academic credentials and for their certification according to the Blockcerts protocol. From such an analysis it results that the Blockchain protocol does not provide any strong mechanism for authenticating the issuing institution, since the issuer authentication is basically performed on the basis of an unauthenticated issuer profile available online and referenced from inside the certificate.

We have shown how a legitimate issuing institution can be easily impersonated by suitably fabricating a fake issuer profile. This way, apparently legitimate academic credentials can be released, which the Blockcerts validation mechanisms are unable to distinguish from valid academic credentials issued by the legitimate institution. This clearly highlights a vulnerability of this protocol, especially when it is used for the certification of academic credentials with legal value.

In order to overcome such a vulnerability, we have proposed to resort to a classic PKI and replace the issuer profile with a certificate signed by a recognized certification authority. This, however, infringes the decentralized nature of the Blockcerts infrastructure. Alternatively, a decentralized identity system could be used instead of a classic PKI to preserve the fully decentralized nature of the paradigm. Such systems, however, are currently under development, and cannot provide an immediate solution to the highlighted vulnerability.

References

[1] T. Lewin, “Dean at M.I.T. resigns, ending a 28-year lie,” *The New York Times*, Apr 2017. [Online]. Available: https://www.nytimes.com/2007/04/27/us/27mit.html

[2] “Mozilla foundation.” [Online]. Available: https://foundation.mozilla.org/en/

[3] “MacArthur foundation.” [Online]. Available: https://www.macfound.org/

[4] “Open badges.” [Online]. Available: https://openbadges.org/

[5] “Diploma supplement.” [Online]. Available: https://ec.europa.eu/education/diploma-supplement_en

[6] A. Grech and A. F. Camilleri, “Blockchain in education,” *JRC Working Papers JRC108255, Joint Research Centre (Seville site)*, 2017. [Online]. Available: https://publications.jrc.ec.europa.eu/repository/bitstream/JRC108255/jrc108255_blockchain_in_education%281%29.pdf

[7] G.-A. Dima, A.-G. Jitariu, C. Pisa, and G. Bianchi, “Scholarium: Supporting identity claims through a permissioned blockchain,” in *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)*, Sep 2018.

[8] R. Arenas and P. Fernandez, “Credenceledger: A permissioned blockchain for verifiable academic credentials,” in *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, Jun 2018.

[9] P. Ocheja, B. Flanagan, H. Ueda, and H. Ogata, “Managing lifelong learning records through blockchain,” *Research and Practice in Technology Enhanced Learning*, vol. 14, no. 4, pp. 1–19, 2019.

[10] A. Mikroyannidis, J. Domingue, M. Bachler, and K. Quick, “Smart blockchain badges for data science education,” in *2018 IEEE Frontiers in Education Conference (FIE)*, Oct 2018, pp. 1–5.

[11] “MIT media lab.” [Online]. Available: https://www.media.mit.edu/

[12] “Learning machine.” [Online]. Available: https://www.learningmachine.com/

[13] “Blockcerts blockchain credentials.” [Online]. Available: https://www.blockcerts.org/

[14] Bestr. CINECA. [Online]. Available: https://bestr.it/

[15] “JSON for linking data.” [Online]. Available: https://json-ld.org/

[16] J. Liu and Z. Liu, “A survey on security verification of blockchain smart contracts,” *IEEE Access*, vol. 7, pp. 77 894–77 904, Jun 2019.

[17] R. Zhang, R. Xue, and L. Liu, “Security and privacy on blockchain,” Aug. 2019, https://arxiv.org/abs/1903.07602.

[18] “Open badges v2.0 IMS final release,” Apr 2018. [Online]. Available: https://www.imsglobal.org/sites/default/files/Badges/OBv2p0Final/index.html

[19] “Open badges infrastructure context file.” [Online]. Available: https://www.imsglobal.org/sites/default/files/Badges/OBv2p0Final/v2/context.json

[20] “Badges and blockcerts,” Jan 2019. [Online]. Available: https://www.learningmachine.com/badges-and-blockcerts/

[21] “blockchain-certificates/cert-tools.” [Online]. Available: https://github.com/blockchain-certificates/cert-tools

[22] “blockchain-certificates/cert-issuer.” [Online]. Available: https://github.com/blockchain-certificates/cert-issuer

[23] “Regulation EU no 910/2014 of the European Parliament and of the Council,” Aug 2014. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32014R0910&from=PL#d1e1772-73-1

[24] CEN/CENELEC Focus Group BDLT, “Recommendations for successful adoption in Europe of emerging technical standards on distributed ledger/blockchain technologies,” Tech. Rep., Jul 2018.