Semantic Notation of Access Control Technology based on eID Identification, FIDO2-Authentication and Attribute-Based Authorization in Digital Environment

Anton Zhidovich and Alexei Lubenko and Iosif Vojteshenko
Belarusian State University
Minsk, Belarus
{anton.zhidovich, alexeilubenko02}@gmail.com, voit@bsu.by

***Abstract*—The paper outlines the semantic OSTIS notation of a user identification system based on ICAO electronic documents, FIDO2 authentication and attribute-based access control. Anonymous credentials, which greatly enhance user privacy, are being considered for implementation in the system. Obtained results will allow to increase the efficiency of the joint use of FIDO2 authentication, modern methods of identification and authorization.**

***Keywords*—FIDO2 technology, ABAC, eID, ICAO passwordless authentication, OSTIS technology, biometrics**

I. INTRODUCTION

A growing trend in information security is the use of passwordless authentication methods, in particular the FIDO2 specification, which enhances the user security and privacy compared to traditional password methods.

It's advisable to use semantic notation to describe the process of setting user identity and privileges based on the joint use of the three components mentioned above. This will increase their level of integration, which is particularly important in the context of multi-agent systems where each component may have its own security system.

In the further presentation the universal language of semantic representation of knowledge in the memory of ostis-systems — SC-code and the language of external graphical representation of SC-code constructions — SCg-code will be used to record knowledge [1].

II. FIDO2 AUTHENTICATION

In [2], [3] a semantic approach to designing systems using FIDO2 authentication was proposed and the benefits of FIDO2 were highlighted.

FIDO2-authentication is a modern, secure and convenient phishing-resistant method based on open standards and implemented in browsers and operating systems. The method provides ease of use by allowing users to register their device with a given online service through the selection of a local authentication mechanism [2].

A. Potential drawbacks of FIDO2 authentication

Despite its significant benefits, FIDO2 authentication has a few of potential drawbacks. Access to the hardware or software token loss can make it difficult to recover an account. In most cases, developers solve this problem by adding an additional authentication method, such as password-based.

Another solution is recovery codes. This approach is often used in various systems outside of FIDO2 authentication and in this case all responsibility for storing passwords rests with the user.

III. ATTRIBUTE-BASED ACCESS CONTROL

A. Definition and key benefits

Attribute-based access control
⇒ *acronym**:
[ABAC]
:= [An authorization model where attributes (or characteristics) are used to determine user permissions]

The main benefits of attribute-based access control include:

1) Possibility of building rules close to domain business terms from the modeled subject area.
2) No restrictions on the complexity of rules, which increases the flexibility of the security system.
3) Possibility of supporting rules with dynamic parameters.
4) Possibility of filtering data the user has access to.

Therefore, attribute-based access control provides the ability to create flexible rules that more closely match the requirements of a particular task, given the context and characteristics of the environment. Thus, ABAC as the basis of an authorization system allows the creation of rules that promote semantic interoperability between security components.

B. Attribute-based access control system elements

Attribute-based access control

$=$ {• _Subject attribute_
- $\Rightarrow$ _synonym*_:
  - [User attribute]
- $:=$ [the characteristics of the user trying to access the resource.]
- $\Rightarrow$ _example*_:
  - {• _identifier_
  - • _departmental affiliation_
  - • _age_
  - }

• _Resource_
- $:=$ [the object the subject is trying to gain access.]
- $\Rightarrow$ _example*_:
  - {• _file_
  - • _application_
  - • _API endpoint_
  - • _server_
  - }
- $\ni$ _Resource attributes_
  - $\Rightarrow$ _example*_:
    - {• _resource creation date_
    - • _resource owner_
    - • _resource type_
    - }

• _Action_
- $:=$ [what the user intends to do with the resource.]
- $\Rightarrow$ _example*_:
  - {• _create, read, update, delete (CRUD)_
  - • _fulfilment_
  - • _replication_
  - }

• _Environment_

$:=$ [the context in which the action request is created and processed.]

$\ni$ _environmental attributes_
- $\Rightarrow$ _example*_:
  - {• _time and location of the action request_
  - • _subject device_
  - • _network protocols used_
  - }

}

C. Usage examples for attribute-based access control

ABAC is widely used in enterprise systems as well as in user applications and IoT devices to improve the security and efficiency of access control to information and resources. It allows a wide range of access problems to be solved with minimal administrative control [4].

Consider examples of ABAC usage:

1) In financial systems to control access to transactions and financial operations.
2) In [5], an ABAC model is proposed to manage access to emergency patient data.
3) ABAC is also used in identity and access management services on such platforms as Amazon Web Services, Google Cloud Platform, Microsoft Asure and Okta.

D. Joint use of FIDO2 and ABAC

The joint use of FIDO2 authentication and attribute-based access control can significantly increase the flexibility and scalability of a security system. It is important to recognise that these are two different processes and that the underlying FIDO2 specification (specifically WebAuthn) does not provide for such integration. Consequently, configuring ABAC and FIDO2 together may require additional effort and resources specific to each system or task.

IV. USE OF ANONYMOUS ATTRIBUTES

By anonymous credentials, we mean a way of implementing attribute-based access control where the matching of user characteristics is proven with Zero Knowledge Proof (ZKP). For example, proving that an age matches a required value without revealing any personal identity.

Anonymous credentials can greatly enhance user privacy, as well as provide authorization with unlinkability. This means that different authorization attempts by a user cannot be linked by a relying party. To achieve this, ZKP is used to prove the user's compliance with the required policy.

Therefore, to implement anonymous credentials in the developed system, it is required that the client (reader) has the software capability to issue zero knowledge proof and the relying party has the capability to verify it.

V. SUBJECT ATTRIBUTES STORAGE

One of the key issues in implementing an attribute-based access control system is the choice of the storage for user attributes. The classic solution is to store this data on the information system side, for example in the database of the authorization service.

In [6] to use the ABAC model in conjunction with the OAuth 2.0 authorization protocol is proposed. This is a logical solution, but in this case there is no way to verify user credentials provided by a service provider such as Google or GitHub. A suitable option is to use electronic documents that store the signature of the issuing party.

A. Electronic identifiers

Let us introduce the definition of electronic identification means in the form of semantic code.

Electronic identification mean
⇒ *synonym**:
[electronic identifier, eID]
:= [an identification document containing up-to-date biographic and biometric information about the owner, the issuer's signature and having built-in data protection mechanisms.]
⇒ *interoperability levels**:
{• *legal*
:= [The use of eIDs is in accordance with government and issuer legal requirements and standards.]
• *organisational*
:= [The use of eID requires adherence to standards and organisational principles, for example in the issuance, implementation and updating of identification tools.]
• *semantical*
:= [eID enables the exchange of structured owner information between participants in a digital environment. This facilitates the establishment of a common meaning and understanding of the data transferred between systems.]
• *technical*
:= [The use of eIDs requires compliance with technical standards such as security specifications, communication protocols and data formats.]
}

One form of eID is an electronic identification card or smart card with an embedded RFID microchip, similar to those used in biometric passports. The chip contains an electronic means of biometric identification with personal data of the biometric document holder in accordance with the requirements of the International Civil Aviation Organisation (ICAO) [7].

In 2005, ICAO's 188 member states approved a new standard requiring all states to begin issuing machine-readable travel documents (MRTDs) in accordance with Doc 9303. The MRTD contains mandatory visual data and separate mandatory summary data in a machine-readable format, as well as a contactless integrated circuit for biometric holder identification [8].

B. eID data structure

To ensure a global level of interoperability and semantic compatibility, ICAO standardized the logical data structure (LDS) that stores information about the electronic identifier holder. It is divided into 16 data groups, each contains specific information about the holder required to identification. (Table I) [9].

Table I: eID Logical Data Structure.

Data Group	Data Element
DG 1	Document Details
DG 2	Encoded Headshot
DG 3	Encoded Face
DG 4	Encoded Fingerprint
DG 5	Encoded Palm print
DG 6	Encoded Iris biometrics
DG 7	Displayed Portrait
DG 8	Reserved for Future Use
DG 9	Signature
DG 10	Data features
DG 11-13	Additional Details
DG 14	CA Public Key
DG 15	AA Public Key
DG 16	Persons to Notify
SOD	Security Data Element

C. eID data protection protocols

To ensure a high level of security and integrity of the data presented on an electronic document, the ICAO standard provides the following security protocols.

Basic Access Control (BAC) is designed to ensure that card data is only accessed when the card is physically present. To do so, a key based on the document number, the holder's date of birth and the document's expiry date is generated. When attempting to access the eID data, the generated key is compared with the stored one, and if they match, access is granted (Figure 1).

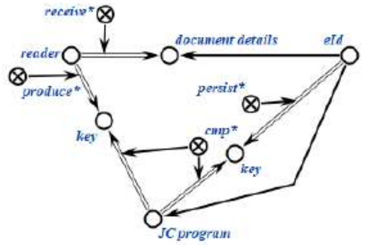

Figure 1: Basic Access Control

Passive Authentication (PA) is a mandatory verification protocol defined in the ICAO standard. Its name refers to the fact that no additional computation is required from the eID chip. The reader retrieves a Document Security Object (DSO) containing a signed hash value of the data. The signature is then verified using the issuer's public key. The hash value of the eID data is also calculated and compared to the value retrieved from the DSO. If these two steps are successfully completed, the passive authentication is passed (Figure 2).

Active Authentication (AA) is designed to protect document data from modification and cloning. It does this by generating a pair of public and private keys when issuing an electronic document. The public key is transmitted with the data to the reader and the private key is stored in the secured eID memory. During active authentication, the received public key is checked against the private key. To do this:

1) The reader sends a cryptographically random string to the eID.
2) The eID signs the string using the stored private key and sends it to the reader.
3) The reader verifies the signature using the received public key. [10] (Figures 2, 3).

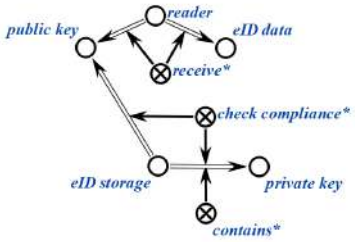

Figure 2: Active Authentication

Chip authentication is an alternative to active authentication, i. e. it allows the terminal (reader) to verify the authenticity of the chip on the electronic document [7].

Electronic documents that comply with the ICAO standard therefore include security mechanisms that allow the authenticity of the document to be verified and prevent the data contained in the document from being stolen or cloned. This standard carefully takes into account modern technological advances and information protection requirements, ensuring a high level of confidence and reliability in the use of documents for personal identification.

VI. PASSWORDLESS AUTHENTICATION WITH ATTRIBUTES AND MEDIATOR (PAwAM)

This chapter considers the implementation of a security system based on identification using ICAO eID, FIDO2 authentication and attribute-based access control.

In [10], a comprehensive and indusrtry-ready solution for the use of anonymous credentials with local or remote attestation is proposed in the form of the FIDO-AC framework, which is an extension of the basic FIDO2 specification. An evaluation of the security and privacy provided by the resulting system and the realisation of a working prototype are also presented.

The essence of the obtained solution is the creation of an additional party — a mediator, which is responsible for the validation of the user data obtained from the eID document.

The FIDO-AC system has the following parties highlighted:

1) The FIDO server (relying party), which forms requests to the FIDO client according to the FIDO2 specification and verifies the response received.
2) The FIDO authenticator used to operate the underlying FIDO2 specification.
3) A client (browser / mobile application with WebAuthn API support) to interact with the FIDO server, the authenticator and the FIDO-AC application.
4) The FIDO-AC application is responsible for reading data from the user's eID. It performs basic access control.
5) The Mediator is responsible for verifying the data received from the eID. It performs active and passive authentication.
6) ICAO eID.

The FIDO-AC application can be specific to each platform and operating system. For example, most modern mobile devices have a built-in NFC sensor with a uniform operating system-level interface, while desktop operating systems have a wide range of different NFC readers from different vendors, the software interaction with which can vary widely. Therefore, Android is proposed as an operating system for FIDO-AC application in [3].

A. *FIDO-AC parties interaction stages*

According to the [10], the work of the FIDO-AC system starts with a user pre-reading the data from the electronic identifier. It is suggested, but not mandatory, to cache the data in the memory of the mobile device for further performance improvements and to reduce the response time of the system.

Then, when attempting to perform an action on an information system resource, the FIDO server generates a signature request that is received by the FIDO client. The request is within the standard WebAuthn specification [11] and contains a cryptographically random bytes buffer — challenge.

On the side of FIDO client, the FIDO-AC application intercepts the request. The FIDO-AC application generates a request containing: eID data hash and signature generated with the issuer's private key for passive authentication; eID public key for active authentication; challenge buffer for unique mediator signature generation. When performing PA and AA, the mediator verifies that the data recorded on the eID card is valid and that the card contains a valid issuer signature.

Once the mediator has successfully verified data 'liveliness', it issues a signature based on the challenge buffer, which also provides additional privacy to the user, as multiple different mediator interactions with the same eID document cannot be linked.

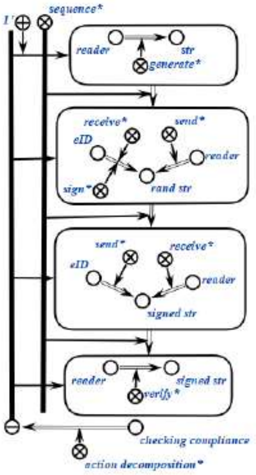

Figure 3: Active Authentication. Checking compliance.

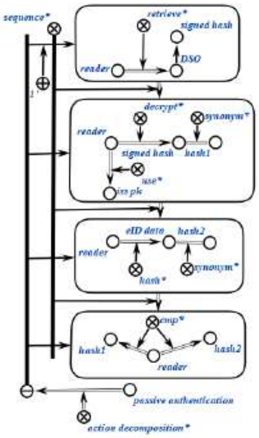

Figure 4: Passive Authentication.

The FIDO-AC application using the ZKP proposed in IV proves, on the one hand, the possession of the subject's attributes and, on the other hand, the conformity of the attributes with the policy required by the FIDO server. The signature issued by the mediator and the proof of the FIDO-AC application are appended to the challenge buffer and sent to the relying party.

At the end of the FIDO-AC process, in contrast to the basic WebAuthn specification, the FIDO server verify not only that the received challenge buffer matches the sent one (except for the attached part), but also the mediator's signature with the proof of FIDO-AC application. After all checks, the FIDO-AC process is considered complete.

The described process can be represented in the form of the SCg-code shown in Figure 3.

B. Implementation options

To implement a FIDO server, it is reasonable to use the ASP.NET Core framework, which has wide deployment and scaling capabilities. As a component allowing the server to act as a relying party, you can use, for example, the FIDO2 component from Rock Solid Knowledge [12].

Web hosting services such as Somee and Google Cloud Platform are offered to test and debug the web application. In addition, most deployment scenarios involve Docker containerisation. In real-world scenarios, the application is deployed on the company's enterprise server.

Currently, all modern desktop and mobile browsers have built-in WebAuthn API support, so any of them can act as a client.

Since FIDO-AC does not require physical separation of the mediator and FIDO-AC application sides, they can be joined together. In this case, there is no need to provide a secure connection between the mediator and the FIDO-AC application.

The Android operating system can be selected as the application platform. In this case, the device must be equipped with a built-in NFC sensor to read the data from the electronic ID.

To achieve correct and proper interaction between an Android device and a biometric document, a JMRTD component can be used [13]. JMRTD provides the ability to connect to the electronic identifier, retrieve data from it and perform the verification described above, supporting ICAO standard documents.

"AVEST" CC, the leading manufacturer of electronic document management security systems in Belarus, provided a functional model of an electronic identifier equivalent to the one used in Belarus as part of the technical assistance for the implementation of this project. It is used for debugging the system and contains the necessary data.

C. Potential vulnerabilities

Let's take a look at the attacks that can be carried out on the mediator / application side of FIDO-AC. One of the key assumptions made by the authors of FIDO-AC is the integrity of the underlying hardware of the device, in our case a mobile phone. This requires the correct operation of the scanner and the trusted platform module, which stores the private keys of the mediator and the platform-dependent authenticator. The latter means that third party access to the device's protected memory is impossible.

The [14] describes that L1 level FIDO2 is vulnerable to timing attacks based on analysing the time taken to execute a cryptographic algorithm depending on the input data. In this way, data such as the private key can be found. A power attack works in a similar way, measuring the power consumed by the device rather than the time taken to complete an operation.

One possible attack is a microarchitecture side-channel attack linked to processor vulnerabilities. For example, a vulnerability in Intel and ARM processors called Meltdown [15] became known some time ago. It exploits

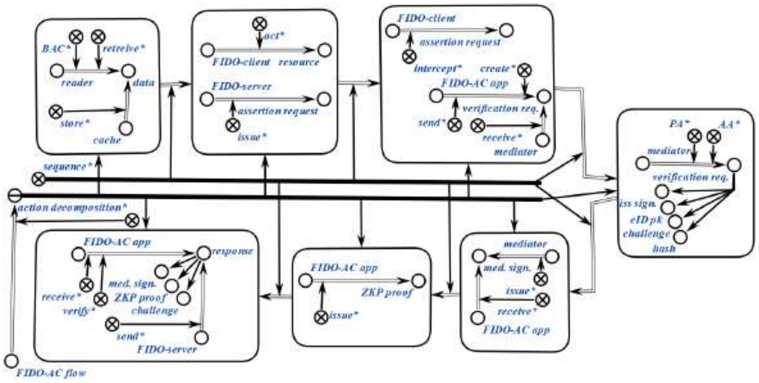

Figure 5: FIDO-AC flow

a bug in the implementation of speculative instruction execution, which causes the processor to ignore page access rights when speculatively executing instructions that read from memory.

In summary, while the mediator side is protected against most software attacks and data interception, it is not protected against side-channel attacks, which are inherently exotic, often require additional equipment, and do not apply to ordinary users of information systems.

Acknowledgment

The authors thank the teams of the Department of Software Engineering of the Belarusian State University for their help and valuable comments as well as “AVEST” CC for technical support for further developments.

References

[1] V. Golenkov, Ed., *Tekhnologiya kompleksnoj podderzhki zhoznennogo cikla semanticheski sovmestimyh intellektual'nyh komp'yuternyh sistem novogo pokoleniya*. Bestprint, Minsk, 2023.

[2] A. Zhidovich, A. Lubenko, I. Vojteshenko, and A. Andrushevich, “Semantic Approach to Designing Applications with Passwordless Authentication According to the FIDO2 Specification,” *Otkrytye semanticheskie tehnologii proektirovanija intellektual'nyh sistem [Open semantic technologies for intelligent systems]*, pp. 311–316, 2023.

[3] A. Zhidovich, A. Lubenko, and I. Vojteshenko, “About Enhanced Access Control using FIDO2 Authentication and Attributes,” *Informacionnie tehnologii i sistemi 2023 [Information Technologies and Systems 2023 (ITS 2023)]*, pp. 88–89, 2023.

[4] (2024, Jan) Company Okta's Blog. [Online]. Available: https://www.okta.com/blog/2020/09/attribute-based-access-control-abac/

[5] M. T. de Oliveira, Y. Verginadis, L. H. Reis, E. Psarra, I. Patiniotakis, and S. D. Olabarriaga, “Ac-abac: Attribute-based access control for electronic medical records during acute care,” *Expert Systems with Applications*, vol. 213, p. 119271, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417422022898

[6] A. Belovodov and O. Laponina, “Ispol'zovanie Upravleniya Dostupom na Osnove Atributov v Protokole OAuth 2.0,” *International Journal of Open Information Technologies*, vol. 11, no. 6, pp. 182–189, 2023.

[7] (2024, Feb) The International Civil Aviation Organization. [Online]. Available: https://www.icao.int

[8] (2024, Feb) Doc 9303. Machine Readable Travel Documents: Part 2. [Online]. Available: https://www.icao.int/publications/pages/publication.aspx?docnum=9303

[9] V. K. Kumar, B. Srinivasan, and P. Narendran, “Efficient Implementation of Electronic Passport Scheme Using Cryptographic Security Along With Multiple Biometrics,” *International Journal of Information Engineering and Electronic Business*, vol. 4, no. 1, pp. 18–24, 2012.

[10] W.-Z. Yeoh, M. Kepkowski, G. Heide, D. Kaafar, and L. Hanzlik, “Fast IDentity online with anonymous credentials (FIDO-AC),” in *32nd USENIX Security Symposium (USENIX Security 23)*. Anaheim, CA: USENIX Association, Aug. 2023, pp. 3029–3046. [Online]. Available: https://www.usenix.org/conference/usenixsecurity23/presentation/yeoh

[11] (2022, Jan) Webauthn Specification. [Online]. Available: https://w3c.github.io/webauthn/

[12] (2023, Dec) FIDO2 for ASP.NET | Unphishable second factor and passwordless authentication for ASP.NET Core. [Online]. Available: https://www.identityserver.com/products/fido2-for-aspnet

[13] (2024, Jan) JMRTD: An Open Source Java Implementation of Machine Readable Travel Documents. [Online]. Available: https://jmrtd.org/

[14] M. Kepkowski, “How not to handle keys: Timing attacks on fido authenticator privacy,” *22nd Privacy Enhancing Technologies Symposium*, pp. 705–726, 2022.

[15] C. Metz and N. Perlroth. The New York Times: Researchers Discover Two Major Flaws in the World's Computers. [Online]. Available: https://www.nytimes.com/2018/01/03/business/computer-flaws.html

СЕМАНТИЧЕСКАЯ НОТАЦИЯ ТЕХНОЛОГИИ УПРАВЛЕНИЯ ДОСТУПОМ НА ОСНОВЕ EID-ИДЕНТИФИКАЦИИ, FIDO2-АУТЕНТИФИКАЦИИ И АТРИБУТИВНОЙ АВТОРИЗАЦИИ В ЦИФРОВОЙ СРЕДЕ

Жидович А. А., Лубенько А. А., Войтешенко И. С.

В работе изложена семантическая OSTIS-нотация системы установки личности пользователя и его привилегий, основанной на идентификации с использованием электронных документов, соответствующих стандартам ICAO, беспарольной FIDO2-аутентификации и управления доступом на основе атрибутов. Рассмотрен вариант реализации в системе анонимных учетных данных, значительно повышающих конфиденциальность пользователя.

Полученные результаты позволят повысть эффективность совместного использования FIDO2-аутентификации с современными методами идентификации и авторизации.

Received 13.03.2024