General Authority of Zakat and Tax (GAZT)

Security Features Implementation Standards to the Governor's Resolution on the Electronic Invoicing Generation Implementing No. () dated 16 Rajab 1442H

Version	Date	Updates
1.0	2021-03-18	Public Consultation

Table of Contents

1   Context

This document contains the security requirements for electronic invoice that taxpayers need to meet to comply with the "E-invoicing" Resolution published by the General Authority of Zakat and Tax.

This document complies with the principles defined by NCDC and NCA, where relevant, to ensure the minimum degree of protection required for national data, systems and networks using cryptographic mechanisms, for civilian and commercial purposes. Those principles are defined in the two published documents:
1. NCA's National Cryptographic Standards (NCS – 1 : 2020)
2. NCDC's Digital Signing Policy (Version 1.1: 2020).

These requirements are based on technical definitions from the following standards
1. ETSI EN 319 132-1: Technical Electronic Signatures and Infrastructures (ESI); XAdES technical digital signatures; Part 1: Building blocks and XAdES baseline technical signatures
2. ETSI EN 319 142-1: Technical Electronic Signatures and Infrastructures (ESI); PAdES technical digital signatures; Part 1: Building blocks and PAdES baseline technical signatures
3. W3C Recommendation: "XML-Signature Syntax and Processing".
4. ETSI EN 319 122-1: "Electronic Signatures and Infrastructures (ESI); CAdES digital signatures; Part 1: Building blocks and CAdES baseline signatures".
5. IETF RFC 5035 (2007): "Enhanced Security Services (ESS) Update: Adding CertID Algorithm Agility".
6. ISO 32000-1: "Document management - Portable document format - Part 1: PDF 1.7".
7. IETF RFC 5652 (2009): "Cryptographic Message Syntax (CMS)".
8. RFC 8705: OAuth 2.0 Mutual-TLS Client Authentication and Certificate-Bound Access Tokens

and are enhanced as per the e-invoicing resolution published. References to electronic signatures in these standards are for technical features in broad common use and are not to be interpreted as relating to Saudi Electronic Transaction Law.

The requirements set out in this document, as per the published resolution, are the **minimum** set of requirements that must be complied with by taxpayers and their Electronic Invoice Generation Solutions

1.1   Document structure

This document is structured as follows:
- Chapter 1 Context
- Chapter 2 Cryptographic Stamp Specifications
- Chapter 3: QR code specification
- Chapter 4: EGS Authentication using OAuth Mutual-TLS

1.2   Audience

The audience for this document is GAZT registered VAT Taxpayers generating VAT invoices and their service providers.

These organisations may be:
- Service providers
- Taxpayers
- Software Developers

More specifically, roles addressed are the following:
- ICT Architects

- ICT Security Specialists
- ICT Developers

1.3   Definitions and Acronyms

These definitions and acronyms describe concepts specific to the Electronic Invoicing implementation and are not to be construed as general legal terms.

- **Technical Certification Authority (CA):** An authorized service that issues Cryptographic Stamp Identifiers or provides other services in this connection and in relation to Cryptographic Stamps.
- **Cryptographic Stamp:** The Cryptographic Stamp is a technical digital signature, and in the context of E Invoicing Implementing Resolution it will be the technical digital signature of the hash of the document. A digital signature is a mathematical scheme for verifying the authenticity of documents. In the context of E Invoicing Implementing Resolution, a valid digital signature, where the prerequisites are satisfied, is evidence for the recipient to believe that the invoice was created by the specified sender, and that the content is has not been altered. Cryptographic Stamps in the context of E Invoicing Implementing Resolution are defined according to the ECDSA standard. Applying the Cryptographic Stamp is referred to as "stamping".
- **CRL:** Certificate Revocation List (CRL) Data structure that enumerates digital certificates that have been invalidated by their issuer prior to when they were scheduled to expire.
- **Digital Certificate:** A Cryptographic Stamp Identifier linking a taxpayer's Invoice Generating Solution unit and a trusted party (GAZT) able to confirm the taxpayer's identity. It is used to establish the identity of an individual, organization, or Web server to which the certificate was issued. As far as this document is concerned, the Digital Certificate identifies the entity applying Cryptographic Stamps on e-invoices. The Cryptographic Stamp Identifier (technically a Digital Certificate) is associated with the signing Key pair used to apply Cryptographic Stamps on e-Invoices, therefore it is also going to be referenced as Cryptographic Stamp Identifier.
- **Elliptic Curve Digital Signature Algorithm (ECDSA):** A Digital Signature Algorithm (DSA) which uses keys derived from elliptic curve cryptography (ECC). While functionally providing the same outcome as other digital signing algorithms, because ECDSA is based on the more efficient elliptic curve cryptography, ECDSA requires smaller keys to provide equivalent security and is therefore more efficient.
- **E-Invoicing platform:** System used to receive and/or clear compliant electronic invoices.
- **Certified Solution:** The solutions used to generate invoices according to the requirements specified in the Governor's Resolution on the Electronic Invoicing Generation Implementing No. () dated 16 Rajab 1442H.
- **ETSI:** ETSI is an independent, not-for-profit, standardization organization in the field of information and communications. ETSI supports the development and testing of global technical standards for ICT-enabled systems, applications and services.
- **Hash:** A hash function is any function that can be used to map data of arbitrary size to fixed-size values called hashes that takes up minimal space. A hash procedure is deterministic—meaning that for a given input value it must always generate the same hash value. It is not possible to derive the original data from a hash; hence, hashing is meant to verify that a file or piece of data hasn't been altered.
- **Key Pair:** A set of mathematically related keys, a public key and a private key, that are used for asymmetric cryptography and are generated in a way that makes it computationally infeasible to derive the private key from knowledge of the public key.
- **OCSP responder:** Online Certificate Status Protocol responder. An online service that responds to Certificate Status Requests and that can issue one of three responses: "Valid", "Invalid", or "Unknown," based on Certificate Revocation Lists or other mechanisms provided to it by Certification Authorities.

- **UUID:** Unified Unique Identification Number, is a 128-bit number used to identify information in computer systems used for E-Invoice Generation. UUIDs generation scheme ensures to a high probability that the generated number is globally unique without the need to check a central database.
- **Public key:** The public component of a pair of cryptographic keys associated with Cryptographic Stamp Identifier used for stamping. In a public key cryptosystem, this key of a user's key pair is publicly known.
- **Private Key:** The secret component of a pair of cryptographic keys associated with the Cryptographic Stamp Identifier used for stamping. In a public key cryptosystem, this key is known only by its user.
- **QR Code ("Quick Response Code"):** A type of matrix barcode, with a pattern of black and white squares that is machine readable by a QR code scanner or the camera of smart devices. For this Resolution a QR code must include basic invoice information specified in this document.
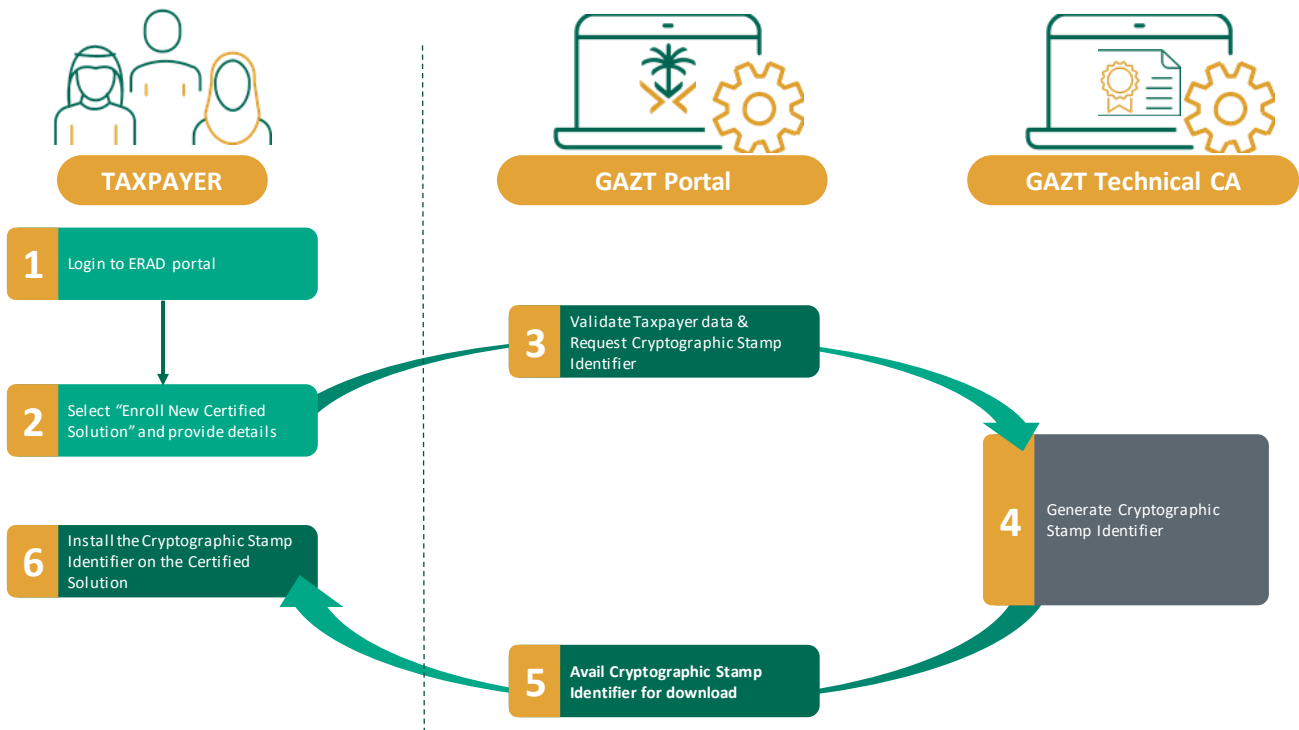
2   Cryptographic Stamp Specifications

2.1   Cryptographic Stamp Business Processes

2.1.1   The processes of Issuance and management of Cryptographic Stamp Identifiers used for Cryptographic Stamping

As part of the EGS onboarding, a digital certificate is going to be issued for the first time, the EGS will store the signing key pair as will as issued certificate in order to use it for stamping e-invoices. The below diagram shows the overall process of enrolling a taxpayer's EGS and issue a digital certificate for it. The process is also described below:

1. The taxpayer representative uses his/her existing credentials to login to GAZT Portal
2. Select "Enroll new EGS" to issue a certificate for an EGS for the first time. Fill In the details required to generate certificate such as:
    a. Device ID, location etc.
    b. A certificate request file generated from the EGS either manually using a tool provided by the EGS vendor or automatically through GAZT Portal
3. GAZT Portal does the necessary business rules validation to before passing the certification request to GAZT's technical CA
4. GAZT's technical CA registers the device into its database and issue a digital certificate then return it back to GAZT Portal
5. GAZT Portal makes the certificate available for download
6. The certificate gets installed on the EGS either manually or automatically through GAZT Portal

Note: Further details on the certificate issuance are going to be published by GAZT as part of the issuing CA business disclosure statement.

2.1.2 Renewal of digital certificates for Cryptographic Stamps

Prior to certificate expiry, the taxpayers will receive a reminder notification from GAZT Portal on the expiry of each certificate. Upon receiving the notification, taxpayers shall follow the below process to renew EGS certificates.

As shown in the diagram, the taxpayer can submit a renewal request as follows:

1. The taxpayer representative uses his/her existing credentials to login to GAZT Portal
2. Select "Renew digital certificate" and provide the certificate and device identification details.
   a. A certificate request file generated from the EGS either manually using a tool provided by the EGS vendor or automatically through GAZT Portal
3. GAZT Portal validates the data entered by the taxpayer then submit the request to the GAZT technical CA
4. The CA validates that the existing certificate is not revoked or renewed before then revoke existing certificate
5. The CA issues a new digital certificate then return it back to GAZT Portal
6. GAZT Portal makes the certificate available for download
7. The certificate gets installed on the EGS either manually or automatically through GAZT Portal
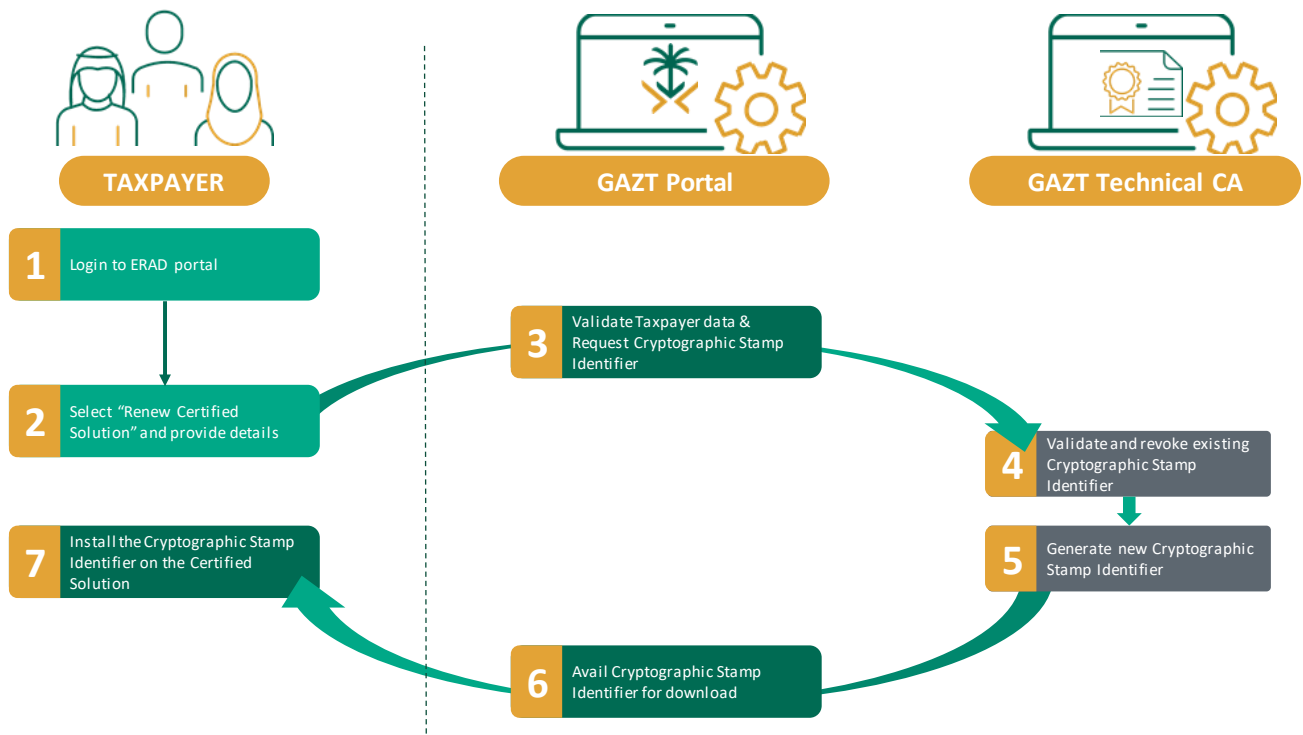

Note: For further details on the certificate renewal please refer to _____ process on GAZT portal.

2.1.3 Revocation of digital certificates for Cryptographic Stamps

The taxpayer needs to revoke the digital certificate of an EGS in the following circumstances in order to avoid any potential unauthorized use of the taxpayer's Cryptographic Stamp:

● If the taxpayer believes that the private key (or the EGS) was stolen or otherwise compromised,

● if the EGS has been damaged, decommissioned or transferred to business unit

● If the taxpayer discovers that the information in the digital certificate is not accurate

As shown in the diagram, the taxpayer can submit a revocation request as follows:

1. The taxpayer representative uses his/her existing credentials to login to GAZT Portal
2. Select "Revoke digital certificate" and provide the certificate and device identification details
3. GAZT Portal validates the data entered by the taxpayer then submit the request to the CA
4. The CA validates that the certificate is valid (not expired or revoked) then revoke the certificate and publish the revocation data
5. Taxpayers and other relevant stakeholders can check the certificate revocation status through the CA publications (CRL/OCSP)
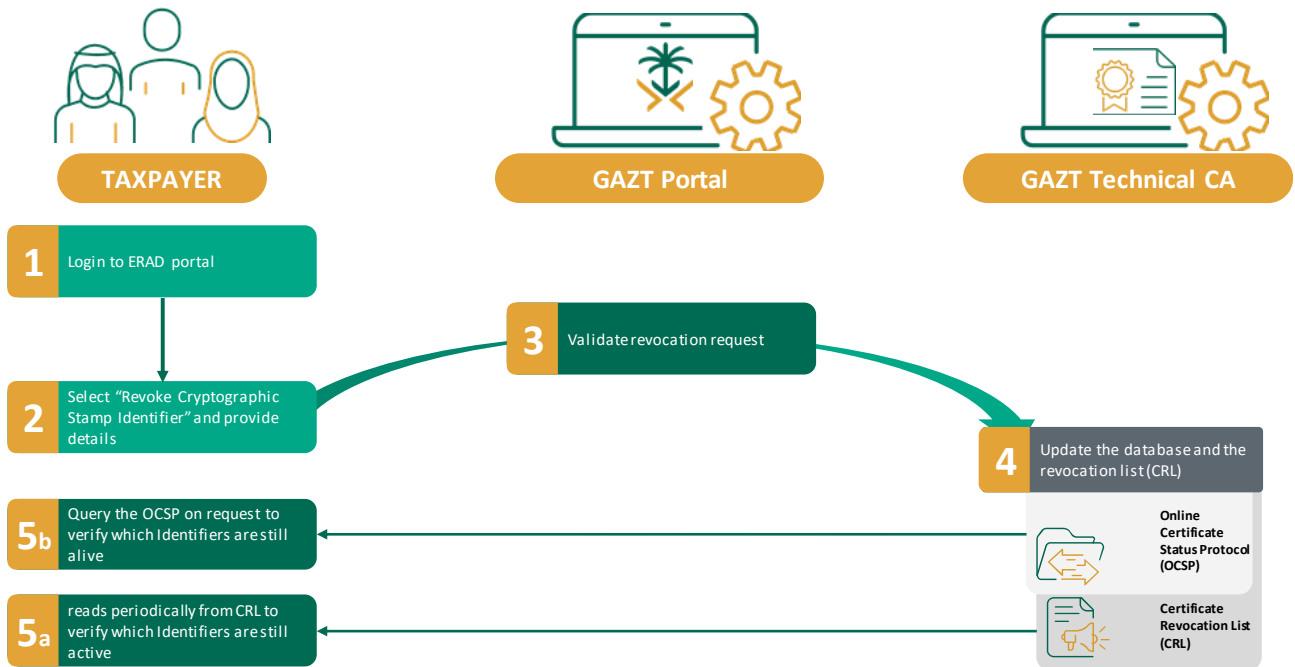
Note: Further details on the certificate revocation are going to be published by GAZT as part of the issuing CA business disclosure statement.

2.2 Standard requirements on the creation and use of Cryptographic Stamps

2.2.1 Requirements

#	Requirement	Description
1.1	Workflow (sequencing and timing) of signatures	**Standard e-invoices:** The below diagram illustrates the invoice generation process at high level where standard e-invoices are stamped by GAZT's centralized e-invoicing platform.

1.2	Workflow (sequencing and timing) of signatures	**Simplified e-invoices:** The below diagram illustrates the invoice generation process at high level where standard e-invoices are stamped by the taxpayer's EGS:
2	Digital certificate issuing CA(s)	GAZT is going to establish its own technical CA for issuing the digital certificates that are going to be used to apply Cryptographic Stamps on e-Invoices.
3	Subscriber agreement	Before using the Digital Stamp Identifier, the taxpayer shall review and agree to the terms and conditions of subscriber agreement with GAZT.
4	Certificate subject	A unique digital certificate is going to be issued to each EGS system belongs to taxpayers as well as to GAZT's e-invoicing platform. Therefore, the certificate is going to include unique identifiers for the system applying Cryptographic Stamps as will entity owning that system such as: ● Taxpayer Identification: VAT registration number

		• EGS Identification: Fully Qualified Domain Names (FQDNs), unique device identifier, IP address or unique system common names, that guarantees uniqueness of device identifier within the domain of each taxpayer Please refer to section B.04 for more details on the information that is going to be stored in the certificate.
5	Accuracy of the data stored in the digital certificate	The digital certificate will store the taxpayer and EGS identification data as mentioned in Req. No. 4. for that GAZT will rely on the data provided by the taxpayer through GAZT Portal without further validation and therefore, the taxpayer is fully responsible for the accuracy and legitimacy of the data provided.
6	key generation	• The Key pair shall be generated according to FIPS 186. Further, reasonable techniques shall be used to validate the suitability of the generated key pair. • The suitability of keys shall be done according to either the ECC Full Public Key Validation Routine or the ECC Partial Public Key Validation Routine. [Source: Sections 5.6.2.3.2 and 5.6.2.3.3, respectively, of NIST SP 56A: Revision 2]. • keys must be marked as non-exportable in order to prohibit key export out of the security module where the key was generated • A hardware or software based security module can be used to generate and store the key pair as long as the above requirements are met
7	Certificate Signing Request (CSR)	The EGS shall be capable of generating a PKCS#10 CSR that includes at least the Certificate CN and Public key. The CSR shall be signed using the private key as a Proof-of-Possession of the private key.
8	Key protection	Taxpayers shall use reasonable techniques to protect their signing key pair (particularly, Private key) and keep it secret be it stored local or centralized. Such techniques include, but are not limited to, disk encryption especially of a software based module is used to store the key.
9	Method of activating private key	Taxpayers are responsible for activating and protecting their signing key. Taxpayers shall use reasonable techniques to secure the activation data of the Private Key that is used to activate the key for signing. Same requirement applies to GAZT's e-invoicing platform.
10	Signature type	For electronic invoices generated in XML format: the Cryptographic Stamp shall follow XAdES digital signatures defined under the ETSI standard [EN 319 132-1]. Please refer to section B.04 for more details on the signature structure and format.

		For electronic invoices generated in PDF/A-3 format (with embedded XML): the Cryptographic Stamp shall follow PAdES digital signatures defined under the ETSI standard [EN 319 142-1]. Please refer to section B.04 for more details on the signature structure and format.
11	Signature packaging	For XAdES, an enveloped signature is required. With enveloped signature, a signature forms a sub-element of the signed XML. For PAdES, enveloped signature that is only supported.
12	Data to be signed	For electronic invoices generated in XML format: the whole XML content except the QR-code data element need to be covered by the signature. For electronic invoices generated in PDF/A-3 format: while the PDF content will be the representation of the XML invoice in a human readable format, the XML invoice itself will still be added as an attachment as specified in ISO 19005-3 titled "Document management - Electronic document file format for long-term preservation - Part 3: Use of ISO 32000-1 with support for embedded files (PDF/A-3)", and contain the compliant XML invoice as an embedded object. In terms of scope/range of the signature, the whole electronic invoice content, meaning the whole document or the whole PDF/A-3 file (including the attached XML invoice), has to be covered by the signature.
13	Time of signing	The time of signing (claimed signing time) is provided based on the clock of the EGS/GAZT e-invoicing platform.
14	Certificate revocation check	Certificate revocation information is published by the CA as CRLs or via OCSP responders. CRLs shall be valid for seven(7) days which would allow EGSs to work fully offline for seven(7) days before it connects to the CRL publication end point in order to download the freshest CRL. Before using the certificate for stamping, EGS/GAZT e-invoicing platform shall check certificate's validity against any of the above mentioned sources.
15	Signature level	For PAdES and XAdES, the signature level should be B-B as Illustrated below. This level incorporate only the elements/qualifying properties that are mandatory, and that implement the mandatory requirements, contain the lowest number of elements/qualifying properties, with the consequent benefits for interoperability.

16	Cryptographic algorithms	• Hashing algorithm shall be SHA-256; • Asymmetric key algorithm shall to ECDSA; • Key length shall be 256.
17	Signature verification	• For the ease of validation by GAZT and other stakeholders, the full chain of certificates from the signing certificate up to GAZT's trust anchor shall be included in the signature. • Certificate path validation and revocation check shall be done as of the time included in the signature. • Signature verification shall be performed according to the ETSI standard [EN 319 102-1] or equivalent.
18	Applicability of the Cryptographic Stamp	The Cryptographic Stamp is intended only to be applied on electronic invoices to establish its authenticity and integrity. As such, GAZT, buyers and potentially other stakeholders are going to verify the stamp to establish the the authenticity and integrity of electronic invoices received from the sellers. GAZT doesn't indorse any other purpose of the Cryptographic Stamp except the use of Digital Stamp Identifier to authenticate the EGS as specified in section 4 EGS Authentication using OAuth Mutual-TLS

2.2.2 Profile specification of the Cryptographic Stamp identifiers

This section describes the X.509 profile of the certificate that is going to be issued by GAZT's CA for stamping e-invoices. The certificate profile complies with X.509 v3 certificates as specified in RFC 5280. While the final certificate profile is going to be published by GAZT in connection with its CA(s) service as part its CP/CPS, the following is provided as an illustrative profile for taxpayers and vendors.

Field / x.509 extension	Value or Value Constant	Field Type/Critical

Version	Version 3	V1 Field
SerialNumber	At least 64 bits of entropy validated on duplicates.	V1 Field
Signature	SHA256 with RSA Encryption	V1 Field
Issuer	<the Subject DN of the issuing CA>	V1 Field
NotBefore	Certificate generation process date/time.	V1 Field
NotAfter	Certificate generation process date/time + Up to 60 months (5 years)	V1 Field
Subject	**For GAZT's e-invoicing system (standard e-invoices):** CN = GAZT centralized e-invoicing system OrganizationIdentifier = <GAZT's 700 number> OU = Standard e-invoices clearance O = GAZT C = SA **For taxpayers' EGSs (simplified e-invoices):** CN = <Fully Qualified Domain Name (FQDN), System unique common name, Public IP address> SerialNumber = <**[Optional]** unique identification code for the EGS> OrganizationIdentifier=<VAT registration number of the taxpayer> OU = <**[Optional]** Taxpayer name branch name> O = <Taxpayer name> C = SA	V1 Field
SubjectPublicKeyInfo	Public Key Key length: P-256	V1 Field
CRL Distribution Points		

Authority Key Identifier	keyIdentifier encoded in compliance to RFC 5280 The keyIdentifier should be composed of the 160-bit SHA-1 hash of the value of the BIT STRING subjectPublicKey of the issuing CA	Extension/NO
Subject Key Identifier	keyIdentifier encoded in compliance to RFC 5280 The keyIdentifier should be composed of the 160-bit SHA-1 hash of the value of the BIT STRING subjectPublicKey	Extension/NO

Certificate Policies	[1]Certificate Policy:
	 Policy Identifier =<OID value to be defined by the CA>
	    [1,1]Policy Qualifier Info:
	      Policy Qualifier Id=CPS
	      Qualifier:
	        <HTTPS URL to the CA repository where the CPS is published>
	
	[2]Certificate Policy **[Optional]**:
	 Policy Identifier =<OID value to be defined by the CA>
		Extension/NO

Authority Information Access	[1]Authority Info Access Method=On-line Certificate Status Protocol (1.3.6.1.5.5.7.48.1) **[Optional]**:
	   Alternative Name: URL=<HTTP URL to the CA OCSP responder>
	
	[2]Authority Info Access Method=Certification Authority Issuer (1.3.6.1.5.5.7.48.2)
	   Alternative Name:
	URL=<HTTP URL to the end point where the CA certificate is published>
		Extension/NO

Key Usage	digitalSignature, keyEncipherment	Extension/YES
Extended Key Usage	clientAuth	Extension/NO

2.3 Structure and Format of the Cryptographic Stamp

2.3.1 Introduction

This section provides a guidance on the required fields and the corresponding values constituting the Cryptographic Stamps for XML and PDF/A-3 (with embedded XML) electronic invoices according to the ETSI standard [EN 319 132-1] and the ETSI standard [EN 319 142-1] respectively.

Note: the guidance provided in this section is meant only to indicate the minimum signature components and any specific values anticipated by GAZT. Hence, the taxpayer/vendors shall follow the detailed standard specifications to ensure full compliance signatures with those aforementioned standards.

Taxpayers are free to choose any commercial off the shelf, open source or even develop their own bespoke tool to generate the stamp according to the aforementioned standards.

2.3.2 Notation for the requirements

The following are key notations on the requirements specified in tables at the rest of this section.

- The column "Elements/Attributes/Services" indicates the name of each component and layout the structure of components where applicable
- The column "Cardinality" indicates the cardinality of the element/attribute/service. Below follows the values indicating the cardinality:
  - 1: exactly one instance
  - 0 or 1: zero or one instance
  - ≥ 0: zero or more instances
  - ≥ 1: one or more instances
- The column "Additional notes and requirements": referencing notes, additional requirements and references to relevant standards

Note: For XAdES, the elements already defined in XMLDSIG [3] appear with the prefix "ds", whereas the new XML elements defined in the present document appear without prefix

2.3.3 Structure of the Cryptographic Stamp in XAdES format

Elements/Attributes/Services	Cardinality	Additional requirements and notes
Signature	1	The Signature element is the root element of an XML Signature. Implementation must generate Signature elements as specified in [1]
1. ds:SignedInfo	1	The SignedInfo structure includes the canonicalization algorithm, a signature algorithm, and the below specified reference URIs.
1.1. ds:CanonicalizationMethod	1	CanonicalizationMethod element specifies the canonicalization algorithm applied to the SignedInfo element prior to performing signature calculations. Implementations must support the required canonicalization algorithms specified in [3].
1.2. ds:SignatureMethod	1	SignatureMethod element specifies the algorithm used for signature generation and validation. This algorithm identifies all cryptographic functions involved in the signature operation (e.g. hashing, public key algorithms, MACs, padding, etc.).
1.3. ds:Reference	≥ 2	Reference element specifies a digest algorithm and digest value, and optionally an identifier of the object being signed and the type of the object. This element shall have at least the below two occurrences in the signature: The first ds:Reference element. Its URI attribute references the data object that has to be signed. ds:DigestMethod indicates the digest algorithm (sha1 in this case) and ds:DigestValue contains the base-64 encoded digest value. The second ds:Reference element. Its URI attribute points to the SignedProperties element (using the URI attribute) that contains the whole set of signed properties. ds:DigestMethod indicates the digest algorithm (sha1 in this case) and ds:DigestValue contains the digest value filtered in base. This means that the digest value of that SignedProperties is included in ds:SignedInfo

		and in consequence signed when this element is signed. This element shall also include the Type attribute with its value set to: "http://uri.etsi.org/01903#SignedProperties."	
1.3.1. ds:Transforms	0 or 1	Transforms element specifies the transformations performed prior to digesting.	
1.4. ds:SignatureValue	1	SignatureValue element contains contains the actual value of the digital signature; it is always encoded using base64 [RFC2045].	
1.5. ds:KeyInfo	1	KeyInfo element contains cryptographic material to verify the signature.	
1.5.1. ds:X509Data	1	X509Data element contains an X509Certificate element for the digital certificate (signing certificates) and all other chain certificates required to build the path up to a trusted anchor.	
1.5.2.1. ds:X509Certificate	≥ 1	X509Certificate element contains a base64-encoded [X509V3] certificate.	
1.6. ds:Object	1	Object element contains three elements with the properties qualifying both the signature and the signed data object.	
1.6.1. QualifyingProperties	1	QualifyingProperties element shall act as a container element for all the	

			qualifying information that is added to an XML signature. The qualifying properties shall be split into qualifying properties that are cryptographically bound to (i.e. signed by) the XML signature, and qualifying properties that are not cryptographically bound to (i.e. not signed by) the XML signature.
1.6.2.1. SignedProperties	1		SignedProperties contains the SignedSignatureProperties element that contains all the signed properties that qualify the signature.
1.6.2.1.1. SignedSignatureProperties	1		SignedSignatureProperties contains all the signed properties that qualify the signature (SigningTime, SigningCertificate, SignaturePolicyIdentifier).
1.6.2.1.1.2. signingTime	1		signingTime contains the value of the signing instant when the signature has been computed.
1.6.2.1.1.2. SigningCertificateV2	1		The SigningCertificateV2 qualifying property shall contain one reference to the signing digital certificate. The SigningCertificateV2 shall also contain references to all the certificates within the signing certificate path, including one reference to the trust anchor. For each certificate, the SigningCertificateV2 qualifying property shall contain a digest value together with a unique identifier of the algorithm that has been used to calculate it. The first reference in SigningCertificateV2 qualifying property shall be the reference of the signing certificate.
1.6.2.1.1.3. SignaturePolicyIdentifier	1		The SignaturePolicyIdentifier qualifying property shall contain either an explicit identifier of a signature policy (this document).

Elements/Attributes/Services	Cardinality	Additional requirements and notes
1.6.2.1.2. SignedDataObjectProperties	1	The SignedDataObjectProperties element shall contain signed qualifying properties that qualify the signed data object.
1.6.2.1.2.1. DataObjectFormat	1	The DataObjectFormat element provides information that describes the format of the signed data object. The mandatory ObjectReference attribute MUST reference the ds:Reference element.
1.6.2.1.2.1.1. MimeType	1	The MimeType element specifies the type of the signed data object. That will be always "text/xml"

2.3.4   Structure of the Cryptographic Stamp in PAdES format

Elements/Attributes/Services	Cardinality	Additional requirements and notes
SignedData.certificates	1	The SignedData.certificates attribute is a collection of certificates including: ● signing digital certificate; ● in order to facilitate path building, generators shall also include all the chain certificates required to build the path up to a trusted anchor.
content-type	1	The content-type attribute indicates the type of the signed content. The content-type attribute shall have value id-data
message-digest	1	The message-digest attribute specifies the message digest of the content being signed.
signature-policy-identifier	0 or 1	The signature-policy-identifier shall contain an explicit identifier of the signature policy (this document).

SERVICE: protection of signing certificate		
SPO: ESS signing-certificate-v2	1	• The signing-certificate-v2 attribute shall be as defined in "ESS Update: Adding CertID Algorithm Agility", IETF RFC 5035 [5], clause 4 "Insert New Section 5.4.1.1 'Certificate Identification Version 2'" • The certHash from ESSCertIDv2 is computed over the entire DER encoded certificate (the signing digital certificate).
Service: provide claimed time of signing		
SPO: entry with the key M in the Signature Dictionary	1	• The time of signing based on the EGS clock. • Refer to ISO 32000-1 [6], clause 12.8.1
entry with key Contents in the Signature Dictionary	1	The Content key shall contain a DER-encoded SignedData object as specified in CMS (IETF RFC 5652 [7]) as the PDF signature. This CMS object forms a CAdES signature described in ETSI EN 319 122-1 [4].
entry with key Filter in the Signature Dictionary	1	• The name of the preferred signature handler to use when validating this signature. if the Prob_Build entrus is not present, it shall be also the name of the signature handler thaw as to create the signature. • A verifier may substitute a different signature handler, other than that specified in Filter, when verifying the signature, as long as it supports the specified SubFilter format. • Refer to ISO 32000-1 [6], clause 12.8.1
entry with key ByteRange in the Signature Dictionary	1	• An array of pairs of intergers (starting byte offset, lenght in bytes) that shall describe the exact byte range for the digest calculation. Refer to ISO 32000-1 [6], clause 12.8.1

		● The ByteRange shall cover the entire file, including the Signature Dictionary but excluding the PDF Signature itself (the entry with key Contents).
entry with key SubFilter in the Signature Dictionary	1	● A name that describes the encoding of the signature value and key information in the signature dictionary. Refer to ISO 32000-1 [6], clause 12.8.1 ● The Signature Dictionary shall contain a value of ETSI.CAdES.detached for the key SubFilter.

3 QR code specifications

3.1 Structure of the QR code

For Electronic Tax Invoices, it is mandatory to generate and print QR code encoded in Base64 format with up to 500 characters that must the fields specified in the below table as per Annex (2) of the Controls, Requirements, Technical Specifications and Procedural Rules for Implementing the Provisions of the E-Invoicing Regulation.

- The QR code fields shall be encoding in Tag-Length-Value (TLV) format with the tag values specified in the "Tag" column of the adjacent table.
- The TLV encoding shall be as follows:
  - Tag: the tag value as mentioned above stored in one byte

[for tags 1 to 5]
  - Length: the length of the byte array resulted from the UTF8 encoding of the field value. The length shall be stored in one byte.
  - Value: the byte array resulted from the UTF8 encoding of the field value.

[for tag 6]
  - Length: length of hash (SHA256) is 32 bytes
  - Value: the byte array constituting the value of each field
- The QR code must also include a Cryptographic Stamp as specified in the next page

Field	Tag	Enforcement date
Seller's name	1	4th June, 2022
VAT registration number of the seller	2	
Time stamp of the invoice (date and time)	3	
Invoice total (with VAT)	4	
VAT total	5	
Hash of XML invoice (or of other format before 1st June 2022)	6	
ECDSA public key *(refer to the next subsection)*	7	
ECDSA signature *(refer to the next subsection)*	8	

3.2 Structure of the Cryptographic Stamp applied on the QR code

The below specifications of the Cryptographic Stamp required for the QR code are based Annex (2) of the Controls, Requirements, Technical Specifications and Procedural Rules for Implementing the Provisions of the E-Invoicing Regulation.

The QR code Cryptographic Stamp shall be generated use the same digital certificate used for stamping the electronic invoices. Please refer to section 3 for the requirements applicable to the management and use of digital certificates used to apply Cryptographic Stamp on electronic invoices.

The Cryptographic Stamp shall be applied on the entire TLV encoding of the QR code fields (from tag 1 to tag 6) then it shall be appended to the TLV structure. The Cryptographic Stamp consists of two fields as shown in the below table:

Field	Tag	Additional requirements and notes
ECDSA public key	7	The public key BLOB format contains only the public portion of an ECDSA key used to generate the Cryptographic Stamp. Length of the public key BLOB for a 256-bit key is 72 bytes
ECDSA signature	8	An ECDSA signature encoded according to IEEE P1363. This signature format encodes the (r, s) tuple as the concatenation of the big-endian representation of r and the big-endian representation of s. Each of these values is encoded using the number of bytes required to encode the maximum integer value in the key's mathematical field. For example, an ECDSA signature from 256-bit elliptic curves (like secp256k1) encodes each of r and s as 32 bytes, and produces a signature output of 64 bytes.

4 EGS Authentication using OAuth Mutual-TLS

GAZT is going to expose different types of APIs during the integration phase, these APIs are going to be used to integrate the taxpayers' EGSs with

GAZT E-invoicing platform for invoice clearance and purposes.

GAZT is going to leverage OAuth 2.0 to secure its APIs, particularly "OAuth 2.0 Mutual TLS Client Authentication and Certificate-Bound Access Tokens" that is specified in RFC 8705 [8] .
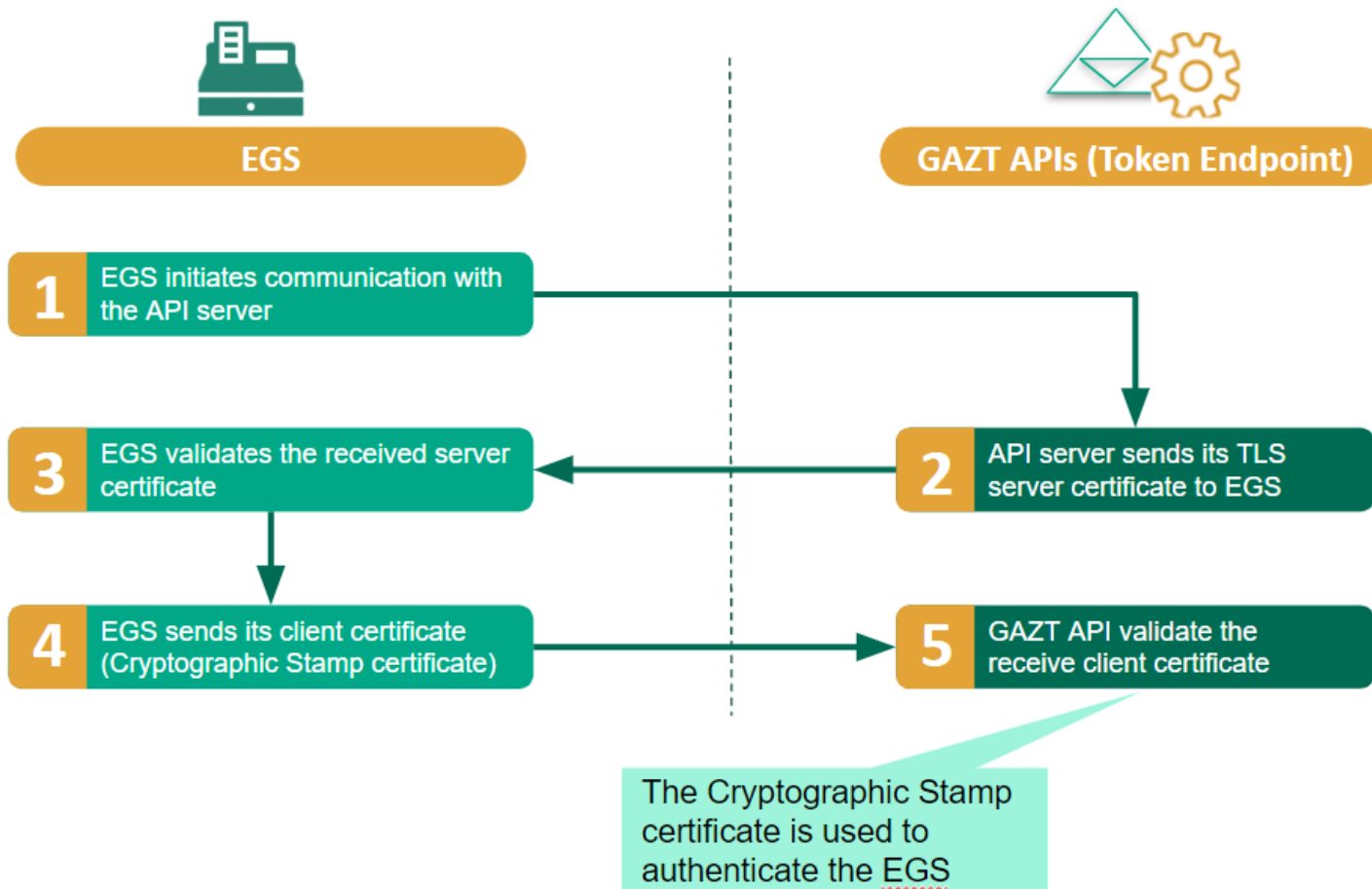
When using mutual TLS the access token provided by the authorization server (GAZT API server) is bound to the client's certificate, this case it is the same digital certificate specified earlier in this document for Cryptographic Stamping.

Mutual TLS certificate-bound access tokens prevent other (unauthorized) clients to re-use the tokens.

The sequence of Mutual-TLS client authentication is depicted by the adjacent diagram (the Server/Client Certificate Verify messages are sent during the TLS Handshake [RFC5246]).

According to section 2.1 "PKI Mutual-TLS Method" in RFC 8705 [8], EGS would be successfully authenticated if:

- the digital certificate path validation was successful as part of the TLS handshake; and
- the subject Distinguished name (DN) in the certificate matches the expected DN configured or registered for that particular EGS

EGS

GAZT APIs (Token Endpoint)

1 EGS initiates communication with the API server

3 EGS validates the received server certificate

2 API server sends its TLS server certificate to EGS

4 EGS sends its client certificate (Cryptographic Stamp certificate)

5 GAZT API validate the receive client certificate

The Cryptographic Stamp certificate is used to authenticate the EGS

Note: More details on the integration and OAuth specifications are published by GAZT. **Please refer to the ___[1] process on GAZT portal.**

---

[1] Process will be published after GAZT platform pilot.