LXI API
Extended Function

Revision 1.0

June 30, 2022

Notices

Notice of Rights All rights reserved. This document is the property of the LXI Consortium. It may be reproduced, unaltered, in whole or in part, provided the LXI copyright notice is retained on every document page.

Notice of Liability The information contained in this document is subject to change without notice. "Preliminary" releases are for specification development and proof-of-concept testing and may not reflect the final "Released" specification.

The LXI Consortium, Inc. makes no warranty of any kind with regard to this material, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The LXI Consortium, Inc. shall not be liable for errors or omissions contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

LXI Standards Documents are developed within the LXI Consortium and LXI Technical Working Groups sponsored by the LXI Consortium Board of Directors. The LXI Consortium develops its standards through a consensus development process modeled after the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Consortium and serve without compensation. While the LXI Consortium administers the process and establishes rules to promote fairness in the consensus development process, the LXI Consortium does not exhaustively evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an LXI Consortium Standard is wholly voluntary. The LXI Consortium and its members disclaim liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other LXI Consortium Standard document.

The LXI Consortium does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. LXI Consortium Standards documents are supplied "as is". The existence of an LXI Consortium Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the LXI Consortium Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every LXI Consortium Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any LXI Consortium Standard.

In publishing and making this document available, the LXI Consortium is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the LXI Consortium undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other LXI Consortium Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

This specification is the property of the LXI Consortium, a Delaware 501c3 corporation, for the use of its members.

Interpretations Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of LXI Consortium, the Consortium will initiate action to prepare appropriate responses. Since LXI Consortium Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, LXI Consortium and the members of its working groups are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. Requests for interpretations of this standard must be sent to mailto:interpretations@lxistandard.org using the form "*Request for Interpretation of an LXI Standard Document*".

This document plus a list of interpretations to this standard are found on the LXI Consortium's Web site: http://www.lxistandard.org

Trademarks Product and company names listed are trademarks or trade names of their respective companies. No investigation has been made of common-law trademark rights in any work.

LXI is a registered trademark of the LXI Consortium

Patents: Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. A holder of such patent rights has filed a copy of the document "*Patent Statement and Licensing Declaration"* with the Consortium. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. Other patent rights may exist for which the LXI Consortium has not received a declaration in the form of the document "*Patent Statement and Licensing Declaration*". The LXI Consortium shall not be held responsible for identifying any or all such patent rights, for conducting inquiries into the legal validity or scope of patent rights, or determining whether any licensing terms or conditions are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility.

Conformance The LXI Consortium draws attention to the document "*LXI Consortium Policy for Certifying Conformance to LXI Consortium Standards*". This document specifies the procedures that must be followed to claim conformance with this standard.

Legal Issues Attention is drawn to the document "*LXI Consortium Trademark and Patent Policies*". This document specifies the requirements that must be met in order to use registered trademarks of the LXI Consortium.

Revision history

Revision	***Description***
1.0 2022-05-10	Initial version
1.0 2022-05-24	This revision updates the term ‘insecure’ to ‘unsecure’ throughout the document.
1.0 2022-06-09	Corrected typo in section 23.10.6.3. The URL of the schema on the LXI web site was inconsistent with other references. Added observation to 23.10.15 pointing out the IDevID cannot be deleted.
1.0 2022-06-16	Corrected the name of LXICertificateRequest/CryptoSuite to LXICertificateRequest/SignatureAlgorithm and specified the error response.

23 LXI API Extended Function

The LXI API Extended Function specifies the methods, semantics, and payload schemas of the LXI REST APIs.

Devices shall not claim compliance with the API Extended Function unless they also comply with an LXI extended function that requires LXI REST API methods. Compliance with the API Extended Function indicates that all of the methods required by other implemented LXI Extended Functions that require REST API methods are implement as specified in this extended function including all the rules and requirements associated with APIs in general and the specifics of the APIs and schemas implemented.

23.1 Purpose and Scope of this Document

The following sections describe the purpose and scope of this specification.

23.1.1 Purpose

This document defines the REST API used by other LXI Extended Functions.

23.1.2 Scope

This document defines a common set of **RULES** and **RECOMMENDATIONS** for constructing a conformant LXI Device with the LXI API Extended Function. Whenever possible these specifications uses existing industry standards.

The original LXI Device Specification included both requirements for all LXI Devices and a number of Extended Functions in a single document. Common information remains in the LXI Device Specification and specific information related to the Extended Function moves to separate documents.

23.2 Definition of Terms

This document contains both normative and informative material. Unless otherwise stated the material in this document shall be considered normative.

NORMATIVE: Normative material shall be considered in determining whether an LXI Device is conformant to this standard. Any section or subsection designated as a RULE or PERMISSION is normative.

INFORMATIVE: Informative material is explanatory and is not considered in determining the conformance of an LXI Device. Any section or subsection designated as RECOMMENDATION, SUGGESTION, or OBSERVATION is informative. Unless otherwise noted examples are informative.

RULE: Rules **SHALL** be followed to ensure compatibility for LAN-based devices. A rule is characterized by the use of the words **SHALL** and **SHALL NOT**. These words are not used for any other purpose other than stating rules.

RECOMMENDATION: Recommendations consist of advice to implementers that will affect the usability of the final device. Discussions of particular hardware to enhance throughput would fall under a recommendation. These should be followed to avoid problems and to obtain optimum performance.

SUGGESTION: A suggestion contains advice that is helpful but not vital. The reader is encouraged to consider the advice before discarding it. Suggestions are included to help the novice designer with areas of design that can be problematic.

PERMISSION: Permissions are included to clarify the areas of the specification that are not specifically prohibited. Permissions reassure the reader that a certain approach is acceptable and will cause no problems. The word **MAY** is reserved for indicating permissions.

OBSERVATION: Observations spell out implications of rules and bring attention to things that might otherwise be overlooked. They also give the rationale behind certain rules, so that the reader understands why the rule must be followed. Any text that appears without heading should be considered as description of the specification.

23.3 Relationship to Other LXI Standards

This specification defines an API that may be required by other LXI Extended Functions. This extended function is only required in conjunction with other extended functions that specify certain API capabilities.

23.4 Acronyms

The following acronyms are used in this specification:

API	Application Program Interface
CMS	Cryptographic Message Syntax, as defined by RFC 8933 or its successors
CSR	Certificate Signing Request
DevID	Device Identifier as defined by IEEE 802.1AR. When used in this document, the clarifications in the LXI Security Extended Function are assumed.
HiSLIP	High-speed LAN Instrument Protocol
HTTP	Hyper-text transfer protocol
HTTPS	Hyper-text transfer protocol performed over a TLS connection
IDevID	Initial Device Identifier as defined by IEEE 802. 1AR. When used in this document, the clarifications in the LXI Security Extended Function are assumed.
LCI	LAN Connection Initialize
LDevID	A locally significant Device Identifier, as defined by IEEE 802.1AR, this is a DevID provisioned to the instrument by the end-customer. When used in this document, the clarifications in the LXI Security Extended Function are assumed.
LXI	LAN Extensions for Instruments
OpenGroup	Standards organization, see OpenGroup.org
PEM	Stands for Privacy Enhanced Mail, although the use in this specification is to refer to the conventional PEM File format for representing X.509 certificates.
REST	Refers to an HTTP API. Stylistically, an API organized as a REpresentational State Transfer API.
SCPI	Standard Commands for Programmable Instruments. Managed by the IVI Consortium.
TCP	Transmission Control Protocol
TLS	Transport Layer Security
VXI-11	VXI-11 specification as managed by the VXI Consortium. VXI-11 provides an ONC RPC-based mechanism for IEEE 488.2 messages.
XML	Extensible Mark-up Language
XSD	XML Schema Definition

23.5 Compliance Requirements

For a device to comply with the LXI API Extended Function, it shall implement the methods required by any other implemented LXI Extended Function. The methods shall implement all endpoints specified for the

method with the semantics, payloads, and headers specified by this extended function document. This includes requirements explicitly called out as rules and any behavior or requirement that states that devices *shall* behave in a certain fashion or provide a certain capability.

The API XML payloads shall comply with the LXI XML schemas. That is, devices shall produce and correctly consume XML documents that are schema-valid based on the LXI schemas. These schema files are not physically included in this document, but are specified in this document and posted at the LXI website as XSD files.

23.5.1 **RULE – Devices Comply with Current Schemas**

The LXI schema's may be updated from time to time. The LXI Conformance Policy indicates the minimum versions devices are required to conform to as part of conformance to a device specification version. Devices shall support schemas that are current at the time of their development, which may be minor revisions more recent than the minimum requirement of the conformance policy.

Devices shall clearly indicate versions of the schema they support.

Devices may also support older schema versions.

23.6 RULE – Include 'LXI API" in the Welcome Web Page "LXI Extended Functions"

Devices implementing the LXI API Extended Function shall include 'LXI API' in the 'LXI Extended Functions' display item of the welcome web page.

23.7 RULE – Include the LXI API Function in the LXI Identification

Devices implementing LXI API Extended Function shall include a <Function> element in the <LxiExtendedFunctions> XML element with the FunctionName attribute of "LXI API" and a Version attribute containing the version number of this document.

For details of the LXI Identification document *Function* element, see: 23.11.4, *LXIExtendedFunctions*.

Example:

<Function FunctionName="LXI API" Version="1.0"/>

23.8 Editorial Conventions Used in This Document

The following conventions are used in this document:

- References to elements or attributes of elements outside of the element/attribute being currently described use italicized XPATH syntax. This syntax represents the element hierarchy much like a file system path. Attributes are indicated with a leading ampersand ('@').
    - For instance, *foo/bar/@baz* refers to the attribute *baz* of the element *bar* which is contained in the element *foo.*
- Elements
    - There are numerous cases where some parent element may be optional, however if the parent element *is* present, then certain child elements or attributes may be required. Thus, if the parent is present, then the child elements and attributes shall be included as specified. In these cases, the parent is listed as optional, and the child elements and attributes are

listed as required. This practice includes both syntactic requirements and LXI standards rules.
    - The requirements for the usage of elements are called out in the requirements column of tables of subelements where that element is referenced.
    - The documentation of the element itself, deals with requirements on the use and syntax of that element, regardless of if it is required in the context of the parent element.
- The specification identifies many elements and attributes that are syntactically optional, that are required to be implemented by LXI devices.
    - The implementation requirements for elements are called out in the tables where they are referenced.
    - The requirements for attributes are called out in the *Description* column in the attributes tables with a paragraph title of *Required*. If the attribute requirement is tagged with a RULE, then the attribute shall be implemented as defined.
- Unless stated otherwise, attributes are read/write. Typically, if an attribute is read-only the LCI column of the tables indicates *Read-only*.
- Statements regarding unsecure mode are rules and indicate required behavior of devices in determining if the device is in unsecure mode. The ultimate determination of the device unsecure mode is based on the requirements in the LXI Security Extended Function. The statements in these tables include minimum conditions to put the device into unsecure mode but are not the complete determination.

23.9 Tables Used in this Document

The Element tables in the following sections are a normative part of this specification. The table headers are as follows:

Element	Element indicates the name of the element. Element names are all single Pascal-case (also known as upper camel case) identifiers. Lengthy element names may be split across multiple lines or have spaces inserted for readability.
Type	Type indicates the type of the element. Types preceded with *xs:* are defined by the XML standards. Types preceded with *lxi:* are defined by LXI.
Cardinality	Cardinality indicates if the element is optional or required, and how many times it may be repeated. *Unbounded* indicates it may be repeated indefinitely.
Requirements	Requirements calls out specific RULES regarding the use of the element.

The Attribute tables in the following sections are a normative part of this specification. The table headers are as follows:

Attribute	Attribute indicates the name of the attribute. Attribute names are all camel case identifiers. Lengthy attribute names may be split across multiple lines or have spaces inserted for readability.
Syntax	Syntax indicates information the data type, cardinality, and default value for the attribute.
LCI	LCI indicates the value assumed by this attribute when the LXI LAN Connection Initialize operation is performed.
Description	Description indicates the attribute semantics. The *Description* column includes a paragraph labelled *Requirement* that states implementation requirements. The *Description* column also includes a paragraph labelled *Unsecure Impact* that indicates the impact of this setting on the device Unsecure Mode.

23.10 The LXI Device API

This section describes the LXI Security API. Subsequent sections describe syntactic and semantic API requirements based on the method parameters (payloads).

23.10.1 RULE – API Client Authentication and Authorization

The LXI Device API endpoints with URLs that begin with /lxi/api shall require that the client be authenticated and authorized and that the communication channel be secure. Thus, these endpoints are only served with HTTPS, that is, HTTP over TLS.

The HTTP GET methods for following LXI Device API endpoints do not require authentication, authorization or encryption and are thus available via HTTP as well as HTTPS:

/lxi/identification
/lxi/common-configuration
/lxi/device-specific-configuration

Observation

Users that do not want this information available to unauthorized clients, or want the communication to be encrypted, can disable the unsecure HTTP LXI API endpoints using the LXI Common Configuration API.

Observation

Clients may attempt to access the API without authorization to determine a suitable type of authorization supported by the device, and the endpoints that require authorization.

23.10.1.1 RULE – API Key Authentication

API clients shall be able to authenticate themselves by providing an HTTP request header that supplies an authentication key. Note that the API Key can always be used to authenticate the user regardless of the device configuration based on the Common Configuration API. The authentication key may be generated by the device, or by the device working in concert with external applications. The authentication key is not generated by the client.

When using API key authentication, the HTTP header *X-API-Key* shall be included with the HTTP request to provide the API key to the device.

The procedure used by the customer to acquire the API key is beyond the scope of LXI. However, devices shall not provide the API key over Ethernet using an unsecure connection.

23.10.1.2 RULE – HTTPS Basic and Digest Authentication

API clients shall be able to authenticate themselves by providing HTTP Basic and optionally Digest authentication per RFC7616/RFC7617 or whatever successors are current when the device is designed. The realm for the LXI API shall be "LXI-API".

Per section 23.10.1.3, *RULE – API Requires Authorization*, authenticated users must also be authorized to use the full API. The users list in the *ClientCredential* element permits users to be designated as authorized.

Observation

RFC7617 specifies headers that shall be included with the HTTP functions to authenticate the client. Specifically, the *Authenticate* header is required.

Observation

Clients may prefer Basic or Digest authentication since it allows the client to choose the password.

23.10.1.3 RULE – API Requires Authorization

The authority of authenticated users shall be verified before they are permitted to change the LXI Security Settings via any Ethernet protocol or interface.

This specification requires two mechanisms by which users may be authorized:

- Authorized users may be specified to the device using the API defined in section 17, *RULE – LXI Common Configuration PUT API*. The user list in the *ClientCredential* element can be used to designated users as authorized using the *APIAccess* attribute. Thus, users presenting the *name* and *password* indicated in the *ClientCredential* are permitted to perform privileged operations.
- Users presenting a valid API Key are authorized.

Other authorization determinations beyond the scope of LXI may be used as well. Such mechanisms must be used to initially authorize a user to use the API.

Observation

Changes may be made to the LXI Security Settings via interfaces other than the Ethernet interface such as USB or the instrument front panel. Those are beyond the scope of LXI but should ensure the client is authorized.

23.10.2 RULE – Additional Means of Authorization

LXI devices are permitted to implement additional means beyond the scope of this specification to authorize the API, however such means shall ensure that clients are fully authenticated and authorized.

23.10.3 RULE – LXI Certificate and CSR GUIDs

Several of the LXI APIs reference either certificates, certificate chains or CSRs using a GUID. The GUID is created and managed by the device and shall be made up of an arbitrary string of alpha-numerics and hyphens.

CSRs may be deleted by the user or, from time-to-time, expire on the device. See section 23.10.16.1, *RULE – Minimum* CSR Retention, for LXI requirements.

The device shall ensure that GUIDs do not replicate under foreseeable circumstances including malicious client actions.

When a certificate is posted to the device it shall receive a new GUID, and the GUID for the corresponding CSR shall not be used again.

Observation

There are numerous algorithms the device could use to generate GUIDs. However, if the REST API operations are in excess of 2 microseconds, a simple incrementing 64-bit unsigned integer will only create duplicate GUIDs every million years.

23.10.4 Common Method Requirements

Common method requirement for the APIs are specified below.

23.10.4.1 **RULE – XML Payloads Comply with LXI Schemas**

LXI provides XSD schemas for each of the LXI APIs that uses an XML payload. Devices shall produce schema-valid XML and accept and properly act on any schema-valid XML.

Numerous requirements regarding the use and interpretation of the schema are included in the following sections regarding the schemas and shall be followed by devices.

23.10.4.2 **RULE – Response and Request headers**

Devices shall return the specified response headers.

Devices shall observe the request headers and ensure that a client presenting request payloads based on the LXI-specified payloads and syntaxes are accepted.

23.10.4.3 **RULE – HTTP Return Codes**

If an operation fails, the device shall return the appropriate HTTP status code as summarized below:

HTTP Response Status Code	Conventional Meaning	Detailed description
400	Bad Request	Something malformed with the request, typically the body of the request is invalid either syntactically or semantically
401	Unauthorized client	Client attempted an operation that requires authentication or authorization that was not suitable
403	Forbidden	The client has not provided necessary authorization
405	Method not allowed	The endpoint does exist, but the HTTP method accessed is not defined.
200	OK	
202	Accepted	Request pending (perhaps reboot required). Devices return the 202 response code to indicate that the request was accepted, but the reconfiguration implicit in the request may not be completed till some future time, perhaps after a reboot.

23.10.4.4 **RULE – LXI Problem Details**

When returning errors, devices shall return information regarding the failure using the LXIProblemDetails XML.

The HTTP Response Header returned with LXI Problem Details shall be ‘Content-Type:application/xml’.

23.10.4.5 **RULE – Operation Pending Response Handling**

If an LXI API returns status 202, that is request pending, it shall return the LXIPendingDetails XML. The pending details permits the client to determine details about pending actions and determine when they are complete.

Devices shall include a response header of: Content-Type: application/xml

The LXIPendingDetails XML includes a URL at which the client can perform an HTTP GET to determine the status of the pending operation. The response from that URL shall either be status 200, OK, or a status of 202, accepted with a new LXIPendingDetails XML.

Observation

> If the device never returns a 202, accepted response, operation pending response handling need not be implemented.

23.10.4.5.1 **RULE – Operations That Require User Action Return Operation Pending**

If an LXI API requires user action, it shall return a status of 202, with the LXIPendingDetails XML without waiting for user intervention.

Observation

> The LXIPendingDetails XML indicates that the device is waiting for user intervention.

23.10.4.5.2 **RULE – Accepted Response URL Expiration**

As long as the operation remains pending each response shall return a status of 202 and an LXIPendingDetails XML. The subsequent responses are permitted to use a different URL, therefore the client must base subsequent GETs on the updated URL.

The returned URL shall remain valid at least until either:

- The client performs a GET on the URL (which may return a fresh LXIPending response) or,
- The client executes another HTTP method that returns a pending status or,
- 1 hour has elapsed or,
- The device is rebooted

If the pending operation requires a reboot to complete, the URL may be invalid after the reboot, however, the device should attempt to provide a URL that will remain valid.

Observation

> The device may not be able to return a URL that works after a reboot because the resolution of the IP address based on either the hostname or static addresses may not be possible before the reboot. For instance, if DHCP has just been turned on, the device may not be able to generate a URL that will work after a reboot.

23.10.5 API Summary

URL	HTTP Method	Summary
/lxi/identification	GET	Returns identity information about the device (and connected devices). This is an unsecure API. Note compliant implementations might not include the Content-Type response header.

/lxi/api/common-configuration OR /lxi/common-configuration	GET	Returns the overall device LXI configuration and capabilities. This API is available both over secure and unsecure connections. See 23.10.8, *RULE – LXI Common Configuration GET API*, for differences between the two endpoints.
lxi/api/common-configuration	PUT	Configures the overall device LXI configuration The network settings managed by this API can usually be applied to all devices in a system.
/lxi/api/device-specific-configuration OR /lxi/device-specific-configuration	GET	Returns device-specific configuration and capabilities. This API is available over both secure and unsecure connections. The two endpoints behave identically.
/lxi/api/device-specific-configuration	PUT	Configures device-specific network settings. The network settings managed by this API are potentially unique to a particular device.
/lxi/api/certificates	GET	Returns a list of certificate GUIDs.
/lxi/api/certificates	POST	Places a PKCS#7 style certificate or certificate chain on the device to use with its LDevID. The certificate must be based on a CSR acquired from the device. The GET response XML has the GUID that is used to identify this certificate.
/lxi/api/certificates/<GUID>	GET	Returns the PKCS#7 certificate, certificate chain, or PKCS#10 CSR identified by <GUID>
/lxi/api/certificates/<GUID>	DEL	Deletes the certificate, certificate chain or CSR identified by <GUID>
/lxi/api/certificates/<GUID>/enabled	PUT/GET	Controls and reads if the designated certificate is used by the device.
/lxi/api/get-csr	GET	Acquires a PKCS#10 CSR from the device based on the request parameters.
/lxi/api/create-certificate	PUT	Tell the device to create a self-signed certificate (also known as an LDevID) based on the request parameters.

23.10.6 XML Schemas for Device APIs

The following sections specify the management of the XML schemas that specify the format of the payloads sent to and from the device with the LXI API.

23.10.6.1 LXI Identification Schema Handling

The LXI Identification schema precedes this specification. Although it is part of the LXI API, the location and management of it is independent of the rules in this section. For details see: 23.10.7, *LXI Identification API.*

23.10.6.2 Schema Version Management

The LXI XML schemas are versioned by providing all versions of each schema in its own directory within a directory named *schemas*. The directory name is the schema name as specified in the document. For instance, all versions of the LXI Common Configuration schema are located in the directory:

schemas/LXICommonConfiguration/

Within this directory, the various versions of the schema have filenames that correspond to the version of the schema. For instance, the 2.3 version of the LXI Common Configuration schema would have the filename *"2.3"*. The reference for this version of the schema would then be:

schemas/LXICommonConfiguration/2.3

23.10.6.3 RULE – Schema location on the device

Devices shall provide schemas for each payload produced or consumed by the device.

The schemas, on a device, shall be located at the device URL from the HTTP(S) server ports that serve the specific API, in the directory *lxi.* Thus, the URL for the 1.0 release of the LXI Common Configuration schema shall be:

http(s)://<device>/lxi/schemas/LXICommonConfiguration/1.0

The schemas are also available on the LXI website in the directory *schemas.* Thus, the URL for the 1.0 release of the LXI Common Configuration schema is:

http(s)://lxistandard.org/schemas/LXICommonConfiguration/1.0

23.10.7 LXI Identification API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/identification	GET	None	---	LXIIdentification.xsd	Content-Type: text/xml

The LXI Identification API is defined by the LXI Device specification. Therefore, it is exempt from the rules in this section.

Clients are not required to authenticate themselves to use this API.

The schema is available from the LXI web site at:
https://www.lxistandard.org/InstrumentIdentification/1.0

23.10.8 RULE – LXI Common Configuration GET API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/api/common-configuration OR	GET	None	---	LXICommonConfiguration	Content-Type: application/xml

lxi/common-configuration					

The LXI Common Configuration GET API returns the overall device LXI configuration. The configuration returned in the XML payload may meaningfully be applied to all devices in a system.

23.10.8.1 RULE – The lxi/common-configuration Endpoint Elides User Lists

The lxi/common-configuration endpoint does not require client authentication, therefore, this response shall elide the user lists used for client authentication and authorization.

23.10.9 RULE – LXI Common Configuration PUT API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/ api/common-configuration	PUT	LXICommonConfiguration	Content-Type: application/xml	none	---

The LXI Common Configuration PUT API configures the common device LXI configuration. The configuration represented by the XML payload may meaningfully be applied to all devices in a system.

23.10.9.1 RULE – Ignore Read-Only Attributes On Write

There are several attributes in the LXI Common Configuration Schema that are read-only, that is, they are returned by the device as part of a GET, but they are not intended for use during a PUT.

If a device receives Read-only attributes on a PUT it shall ignore them, and not treat them as an error.

Observation

Ignoring Read-Only attributes simplifies customer use by enabling the user to read the LXI Common Configuration from a device, then subsequently send it back to the device.

23.10.10 RULE – LXI Device Specific Configuration GET API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/ api/device-specific-configuration OR	GET	none	---	LXIDeviceSpecificConfiguration	Content-Type: application/xml

/lxi/device-specific-configuration					

The LXI Device Specific Configuration GET API returns device-specific configuration and capabilities as specified in the LXI Device Specific Configuration schema.

The settings returned by this API are either potentially unique to a particular device or automatically configured.

The two endpoints return the same response.

23.10.11 RULE – LXI Device Specific Configuration PUT API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/api/device-specific-configuration	PUT	LXIDeviceSpecificConfiguration	Content-Type: application/xml	none	---

The LXI Device Specific Configuration PUT API configures network settings that are device-specific or potentially automatically configured.

Devices retain the LXI Device Specific configuration and only utilize it when automatic configuration is disabled. Thus, writing the LXI Device Specific configuration while automatic configuration is active then subsequently disabling automatic configuration will result in the device using the configuration specified in the LXI Device Specific configuration.

23.10.12 RULE – LXI Certificates GET API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/api/certificates	GET	none	---	LXICertificateList	Content-Type: application/xml

The LXI Certificates GET API returns a listing of certificates, certificate chains, and outstanding CSRs on the device. This listing includes information specified in the LXICertificateList schema including GUIDs that identify each entity. These GUIDs may be used, for instance, to designate the LXI Certificate to the DEL method.

CSRs may be deleted by the user or, from time-to-time, expire on the device. See section 23.10.16.1, *RULE – Minimum* CSR Retention, for LXI requirements.

23.10.13 RULE – LXI Certificates POST API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/ api/certificates	POST	CMS(RFC 8933)	Content-Type: application/cms	LXICertificateRef	Content-Type: application/xml

The LXI Certificates POST API provisions a certificate or certificate chain to the device to be used by the device to identify itself. The posted value is a PKCS#7 style certificate or certificate chain for the device to use with its LDevID.

The certificate must be based on a CSR acquired from the device. The CSR is deleted when the new certificate is successfully posted to the device.

The device response XML contains the GUID that may be used subsequently to identify this certificate or certificate chain for use with other APIs.

23.10.14 RULE – LXI Certificate GET API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/ api/certificates/<GUID>	GET	none	---	For certificates or certificate chains: CMS (RFC 8933) For CSRs: PEM (RFC 5967)	For certificates or certificate chains: Content-Type: application/cms For CSRs: Content-Type: application/pkcs10 Content-Transfer-Encoding: base64

The LXI Certificates/<GUID> GET API returns the certificate, certificate chain, or CSR identified by the <GUID> incorporated into the URL. Note that the type of the response is dependent on the GUID.

23.10.15 RULE – LXI Certificate DELETE API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/ api/certificates/<GUID>	DEL	none	---	none	---

The LXI Certificates/<GUID> DELETE API deletes the certificate, certificate chain, or CSR identified by the <GUID> incorporated into the URL.

Observation

The IDevID cannot be deleted.

23.10.16 RULE – LXI CSR GET API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/api/get-csr	GET	LXICertificateRequest	Content-Type: application/xml	PEM (RFC 5967)	Content-Type: application/pkcs10 Content-Transfer-Encoding: base64

The LXI CSR GET API acquires a PKCS#10 CSR from the device. The CSR is created based on the data in the LXICertificateRequest XML which includes the subject and other fields the client specifies for the CSR.

23.10.16.1 RULE – Minimum CSR Retention

Devices shall at least retain the most recently generated CSR for any given cryptography suite at least until a power cycle. Devices should retain CSRs longer than this to support other customer use models, especially those that require operator intervention.

23.10.17 RULE – LXI Create Certificate API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/api/create-certificate	PUT	LXICertificateRequest	Content-Type: application/xml	LXICertificateRef/1.0	Content-Type: application/xml

In response to this call, the device shall create a new certificate (that is, an LDevID) to use to authenticate itself. This self-signed certificate shall be managed and presented to clients consistent with the requirements in the LXI Security Extended Function.

If the device is unable to respect any of the fields specified in the *LXICertificateRequest*, the device shall return an error.

23.10.18 RULE – LXI Certificate ENABLED API

URL	Method	Request Content	HTTP Request Headers	Response Content	HTTP Response Headers
/lxi/api/certificates/<GUID>/enabled	PUT	LXILiterals	Content-Type: application/xml	None	---

/lxi/ api/certificates/<GUID>/enabled	GET	None	---	LXILiterals	Content-Type: application/xml

The LXI Certificates/<GUID>/enabled PUT API enables or disables the designated certificate or certificate chain identified by the <GUID> incorporated into the URL.

23.10.18.1 RULE – LXILiterals Parameter to Enabled is Boolean

The LXILiterals schema permits arbitrarily typed attributes. The request LXILiterals parameter to enabled shall be an attribute of name *value* and of type *xs:boolean*. The value of the Boolean attribute indicates if the certificate or certificate chain identified by the <GUID> is enabled.

23.10.18.2 RULE – LXILiterals Response to Enabled is Boolean

The LXILiterals schema permits arbitrarily typed attributes. The response LXILiterals parameter to enabled shall be an attribute of name *value* and of type *xs:boolean*. The value of the Boolean attribute indicates if the certificate or certificate chain identified by the <GUID> is enabled.

23.11 LXI Instrument Identification Schema

The LXI Identification schema represents LXI device identity information, and identity information for devices that are connected to an LXI device.

23.11.1 Schema Organization and Overview

Figure 1 *The LXI Identification Schema* shows the overall schema graphically.

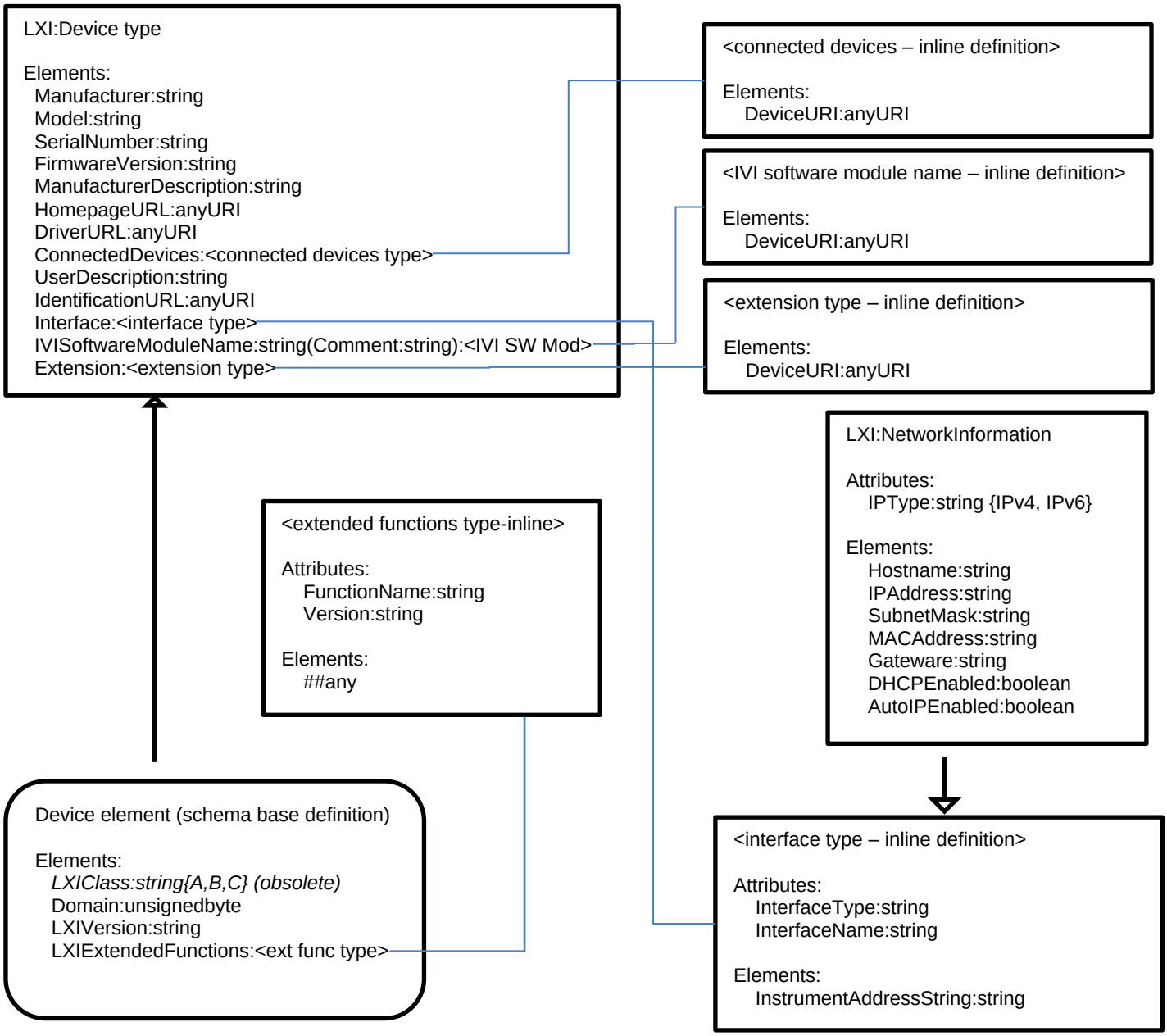

Figure 1 The LXI Identification Schema

The base element that every valid XML must contain is shown in the box with rounded corners. Boxes with square corners show types that are defined in the schema. Note that most types defined in the schema are defined in-line and cannot be individually referenced. The only types that are defined externally are the *lxi:device* complex type and the *lxi:NetworkInformation* complex types.

All of the attribute types in the LXI Identification XML are defined in the XSD schema definition, none are defined by LXI.

The element types that are defined by XSD are shown in the diagram after the tag name of the element. Where types are defined in the LXI Identification schema, they are shown in a separate box, and associated with the element that uses the type with a line.

To retain simplicity in the diagram, the cardinalities of the elements and attributes are not included, but this information is included in the reference sections below.

Thus, to expand the schema into a valid XML, first consider the root Device element (the box with rounded corners). Since the Device element is a generalization of the *lxi:Device* type, valid XML begins with the elements defined in the Device type, followed by the elements added to the type in the Device element. So, valid XML has the elements:

Manufacturer
Model
SerialNumber
FirmwareRevision
ManufacturerDescription
HomepageURL
DriverURL
ConnectedDevices
UserDescription
IdentificationURL
Interface
IVISoftwareModuleName
Extension
<the obsolete LXI Class is placed here in earlier versions>
Domain
LXIVersion
LXIExtendedFunctions

The illustration shows some of the characteristics of each of these elements. The following sections call out the semantics and cardinalities of each element and attribute in detail.

23.11.2 Device Element (the Root Schema Element)

As described in the previous section, the Identification schema begins with a device element that specifies the following additional elements in addition to the base Device type.

Element	Type	Cardinality	Requirements
LXIClass	xs:string, restricted to {'A','B','C'}	Optional	Deprecated in LXI version 1.4. This element indicates if this device is Class A, B, or C.

Domain	xs:unsignedByte	Optional unbounded	The LXI domain(s) this instrument uses for LXI Event Messages. Per LXI Standard 1.6, 4.3 RULE - LXI Event Message Format.
LXIVersion	xs:string	Required	Indicates the latest version of the LXI device specification this device complies with.
LXIExtended Functions	lxi:LXIExtendedFunctions	Optional	LXI Extended functions used to describe extended capabilities of the instrument.

23.11.3 The Device Type

The LXI Device element has information about a general device. The Device type can be used to describe both LXI ethernet devices and devices that may be connected via other interfaces such as GPIB or USB.

LXIDevice is composed of the elements listed in the following table.

23.11.3.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
Manufacturer	xs:string		Manufacturer, should match the manufacturer field in IEEE 488.2 identity query. "Manufacturer" field Per LXI Standard 1.6, 9.2 RULE - Welcome Web Page Display Items.
Model	xs:string		Instrument model designation, should match the model field in IEEE 488.2 identity query. "Model" field per LXI Standard 1.6, 9.2 RULE - Welcome Web Page Display Items.
SerialNumber	xs:string		Instrument serial number, should match the serial number field in IEEE 488.2 identity query. "Serial Number" field per LXI Standard 1.6, 9.2 RULE - Welcome Web Page Display Items.
Firmware Revision	xs:string		Instrument firmware revision, should match the firmware revision field in IEEE 488.2 identity query. "Firmware and/or Software Revision" field per LXI Standard 1.6, 9.2 RULE - Welcome Web Page Display Items.
Manufacturer Description	xs:string		This is the manufacturers product description such as "Acme 1234A Digital Foozywachit".
HomepageURL	xs:anyURI		This is the URL for the instrument Manufacturer.
DriverURL	xs:anyURI		This is the URL where users can go to acquire the latest driver

Connected Devices	lxi:ConnectedDevices		This optional element is used by gateways to advertise information for connected devices, such as GPIB, VXI, USB, PXI, and/or Serial instruments. Per LXI Standard 1.4, 10.2.4 RULE - Connected Device URLs
User Description	xs:string		This is a user description for this device, for instance "the Spectrum Analyzer on the Blue Portable Cart". "Description" field per LXI Standard 1.4, 9.2 RULE - Welcome Web Page Display Items.
Identification URL	xs:anyURI		This is the URL from which the instrument will source the identification XML. Per LXI Standard 1.4, 10.2 RULE - XML Identification Document URL.
Interface	lxi:NetworkInformation	Optional unbounded	Interface Information. For instances of LXIDevice, at least one Interface of type "NetworkInformation" must be provided, with an InterfaceType of "LXI".
IVISoftware Module Name		Optional unbounded	This identifies the IVI driver as specified in the IVI Configuration Store Name field of the Software Module. See Section 2.5.3 IVI Session and IVI Driver Session, in IVI-3.5: Configuration Server Specification. The Comment annotation is used to describe this software module, especially where the driver supports multiple software modules, that is, instrument personalities.
Extension			Vendor specific extensions used to describe the instrument.

23.11.4 *LXIExtendedFunctions*

The LXIExtendedFunctions type contains a list of the extended functions implemented by this device.

LXIExtendedFunctions is composed of function elements each with the FunctionName and Version attribute described in the table below.

Element	Type	Cardinality	Requirements
Function		Optional unbounded	Indicates that an LXI extended function is available, the version of the extended function implemented, and the options associated with it. The Function element contains arbitrary elements.

The Function element has the following attributes.

23.11.4.1 *Function Attributes*

Attribute	Syntax	LCI	Description
FunctionName	**Type:** xs:string **Card.:** Req. **Default:** NA		**The LXI Extended Function Name**
Version	**Type:** xs:string **Card.:** Req. **Default:** NA		**The version of the LXI Extended Function that the device complies with.**

23.11.5 *NetworkInformation*

The network Information element has general information relevant for a network device.

NetworkInformation is composed of the attributes and elements listed in the following tables.

23.11.5.1 *Attributes*

Attribute	Syntax	LCI	Description
InterfaceType	**Type:** xs:string **Card.:** Req. **Default:** NA		For LXI devices, this field must contain LXI. This may be used to designate other vendor specified interfaces (e.g., VXI, PXI, GPIB, Serial, USB, etc.).
InterfaceName	**Type:** xs:string **Card.:** Opt. **Default:** *None*		This field should contain a logical name for the interface, from the device's perspective. For example, network interfaces may be named "eth0", "eth1", etc.
IPType	**Type:** **Card.:** Opt. **Default:** *None*		Identifies the IP implementation as either IPv4 or IPv6.

23.11.5.2 *Sub-elements*

The following must occur in this order:

Element	Type	Cardinality	Requirements
Instrument Address String	xs:string	Optional unbounded	This is a VISA-like string to help the driver or a human determine the actual address string. Consistent with the web presentation of an IVI I/O Resource Descriptor string, per LXI Standard 1.6, 9.2.1 RULE - Instrument Address String on Welcome Page.

Hostname	xs:string	This is the hostname used for DNS. "Hostname" field per LXI Standard 1.6, 9.2 RULE - Welcome Web Page Display Items.
IPAddress	xs:string	This is the currently active IP Address. This is represented as a string and can represent IPv4 or IPv6 addresses. "TCP/IP Address" field per LXI Standard 1.6, 9.2 RULE â€“ Welcome Web Page Display Items.
SubnetMask	xs:string	The currently configured subnet mask. "Subnet mask" field per LXI Standard 1.6, 9.5 RULE - LAN Configuration Web Page Contents.
MACAddress	xs:string	This is the MAC address of this interface. "MAC Address" field per LXI Standard 1.6, 9.2 RULE - Welcome Web Page Display Items.
Gateway	xs:string	The IP address of the currently configured gateway. "Default Gateway" field per LXI Standard 1.6, 9.5 RULE - LAN Configuration Web Page Contents.
DHCPEnabled	xs:boolean	Indicates if the instrument is configured to accept configuration from DHCP, per LXI Standard 1.6, 8.6.1 RULE - Options for LAN configuration.
AutoIPEnabled	xs:boolean	Indicates if the instrument is configured to use AutoIP to choose an IP address, per LXI Standard 1.6, 8.6.1 RULE - Options for LAN configuration.

23.11.6 *ConnectedDevices*

ConnectedDevices contains a list of connected devices, that is, devices that are connected through the primary LXI device.

The *DeviceURI* subelement indicates the URI for child devices.

*The ConnectedDevices complex type has **no attributes***

23.11.6.1 *Sub-elements*

The following must occur in this order:

Element	Type	Cardinality	Requirements
DeviceURI	xs:anyURI	Optional unbounded	URIs for connected devices represent the base URL for the connected device. Per LXI Standard 1.6, 10.2.4 RULE - Connected Device URLs

23.12 LXI Common Configuration Schema

The LXI Common Configuration XML schema is specified by the LXI Consortium as part of the LXI Security Extended Function.

LXICommonConfiguration contains settings related to the device secure configuration. This includes the configuration of the network interface, configuration of various network protocols and client authentication information.

This schema is used to:

- Configure the security settings of a device
- Interrogate a device to determine its security settings
- Interrogate a device to determine its security capabilities

RULE:23.12-1 On an HTTP PUT the device shall go to the state specified in the XML.

A device is configured by performing an HTTP PUT of this XML to the LXI-specified endpoint in the device. A successful PUT indicates that the device recognizes the XML and that it will assume the configuration specified in the XML. The point in time when the new configuration takes effect is device dependent.

If any part of the XML is syntactically invalid or if the XML represents settings that the device does not support, the device state shall report an HTTP error and not change state.

Per the LXI API Specification, the reason for the error shall be elaborated using the *LXIProblemDetails* response schema.

OBSERVATION: *The LXICommonConfiguration/@strict attribute explicitly permits devices to not act on configuration of the listed protocols if they are not implemented. Thus, an XML that includes the configuration of an unimplemented protocol is not an error when strict is false.*

RULE:23.12-2 The device GET response shall indicate the current state and capabilities of the device.

To interrogate a device to determine its settings and capabilities, the client performs an HTTP GET to the LXI-specified endpoint. The device shall reply with an instance of this XML document that reflects the configuration of the instrument.

To determine the capability of the instrument, the client can inspect the XML. Where optional elements are returned (regardless of if they are enabled or disabled), the device is capable of enabling the corresponding configuration. For instance, if a device returns a SCPITLS element, regardless of it is disabled or enabled, the device provides a SCPITLS interface that may be subsequently enabled. If the device does not provide a SCPITLS interface, the optional SCPITLS element shall not be included in the device response.

RULE:23.12-3 Devices shall indicate capabilities not apparent from the queried settings using the capability attribute.

In some cases, devices may be capable of variations on a capability, such as multiple instances on different ports. For those cases, devices shall include the *capability* attribute in the response. The *capability* attribute indicates the variations that can be configured, for instance the names of instances that can be created. When the *capability* attribute is included in the definition of an element, its use is described.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXICommonConfiguration/1.0, version: *1.0*
Editorial date:June 30, 2022

23.12.1 LXICommonConfiguration

LXICommonConfiguration is the root element for the LXI common configuration. It represents the configuration of one or more LXI physical interfaces and user authentication.

The configuration in *LXICommonConfiguration* is generally common to all devices in a system.

23.12.1.1 Attributes

Attribute	Syntax		LCI	Description
strict	**Type:**	xs:boolean	Write-only	*strict* indicates that designated portions of this XML document may not be ignored by the device. This requirement does not bear on attributes and elements that
	Card.:	Opt.		
	Default:	false		

				are explicitly documented to be ignored, for instance, extension attributes. If *strict* is false, devices shall ignore configuration of the following if they are not implemented by the device: • */LXICommonConfiguration/Network/IPv6* • */LXICommonConfiguration/HTTP* • */LXICommonConfiguration/SCPIRaw* • */LXICommonConfiguration/SCPITLS* • */LXICommonConfiguration/Telnet* • */LXICommonConfiguration/HiSLIP* • */LXICommonConfiguration/VXI11* **Required: RULE:23.12.1.1-1** **Unsecure impact:** NA
HSMPresent	**Type:** xs:boolean **Card.:** Req. **Default:** NA		Read-only	*HSMPresent* indicates if the device has a hardware security module. True indicates the device has hardware that ensures that the private keys used to encrypt communication are not stored by the device unless encrypted by a hardware security module that performs encryption of private keys and other secrets using encryption keys that are not visible external from the hardware security module. False indicates the device does not have hardware assistance to protect private keys. **RULE:23.12.1.1-2** *HSMPresent* is a read-only attribute that is true if and only if the device uses a HSM to protect the private keys used for LXI communication. **Required: RULE:23.12.1.1-3** **Unsecure impact:** NA

23.12.1.2 **Sub-elements**

The following must occur in this order:

Element	Type	Cardinality	Requirements
Interface	lxi:Interface	Required unbounded	**RULE:23.12.1.2-1** Devices shall accept configuration based on an *Interface* element for any LXI conformant interface. At least one instance of the Interface element is required. The device shall support a PUT that includes an *Interface* element for any or all interfaces that are LXI conformant.

			RULE:23.12.1.2-2 Devices shall return an *Interface* element for each interface. **OBSERVATION:** *Interfaces that do not conform with LXI specifications only require the Interface/@Enable attribute.* Devices may optionally configure non-LXI conformant interfaces with the Interface element.
Client Authentication	lxi:ClientAuthentication	Optional	**RULE:23.12.1.2-3** *ClientAuthentication* shall be optionally accepted for PUT. **RULE:23.12.1.2-4** *ClientAuthentication* without the *@passwords* or *@APIAccess* attributes shall be returned for GET over secure connections and elided for unsecure connections.

23.12.2 Interface

Interface specifies the settings associated with a single device interface including the common aspects of the ethernet configuration and configuration of protocols served on that interface.

LXICommonConfiguration can represent the configuration of multiple interfaces on a single device, however, devices that implement multiple interfaces shall allow any subset, or all, of the interfaces to be configured using a single XML document. Any LXI Security conformant interfaces on a device shall permit any interface that complies with LXI Security to be configured using this element.

RULE:23.12.2-1 Non-LXI interfaces can be disabled using the *Interface/@enabled* attribute.

RULE:23.12.2-2 Device network interfaces (including those added dynamically) over which the LXI device may be controlled that are not LXI Conformant shall at least support this element with the enabled attribute so that network interfaces that are not LXI Security capable can be disabled.

RULE:23.12.2-3 If any unsecure interface is enabled, then the device shall report that it is unsecure mode.

RULE:23.12.2-4 Absence of optional elements disables them.

If an optional element is absent, the device behavior shall be equivalent to including the element and specifying the enabled attribute of that element to be false. That is, if an element is absent, the capability is disabled.

OBSERVATION: *All optional elements have an enabled attribute as required to implement this RULE.*

RULE:23.12.2-5 If a device does not implement a capability configured by an XML element within *Interface*, it shall omit that optional XML element from its response. If the device does implement the capability, it shall include the element in the response and indicate the current configuration. See the details regarding the implementation of *LXICommonConfiguration/@strict* attribute regarding how certain protocol configurations are handled.

23.12.2.1 Attributes

Attribute	Syntax		LCI	Description
name	**Type:** **Card.:** **Default:**	xs:string Opt. LXI	NA	*name* identifies this physical network interface within the device. It differentiates the interfaces in devices that have multiple interfaces.

				If *name* is omitted, the behaviors is the same as if the name were included with the value "LXI". Some settings may be coupled between interfaces. That is the settings on separate physical interfaces may be required to be the same. **RULE:23.12.2.1-1** Devices with a single interface shall treat the *Interface* element with the name "LXI" (the default name) to configure the single interface. Devices with multiple interfaces shall assign one of them the name "LXI" (the default name). **OBSERVATION:** *Devices may have multiple network interface and choose to configure them all identically and bridge traffic internally.* **Required: RULE:23.12.2.1-2** **Unsecure impact:** NA
LXIConformant	**Type:** xs:string **Card.:** Opt. **Default:** *None*		Read-only	*LXIConformant* is a read-only attribute that indicates the LXI specifications this device complies with. If this interface does not comply with the LXI Device specification, an empty string is used. The returned string is a comma separated list of the LXI specifications this interface complies with. The individual substrings are the same as those defined for the Identification schema. **OBSERVATION:** *Clients should ignore white space in these strings.* **OBSERVATION:** *An empty string indicates that the interface is not LXI compliant. However, an interface that does not comply with LXI is not required to implement this attribute.* **Required: RULE:23.12.2.1-3** (read-only) **Unsecure impact:** NA
enabled	**Type:** xs:boolean **Card.:** Opt. **Default:** true		**RULE:23.12.2.1-4** At LCI, all LXI conformant interfaces shall be enabled, others may be enabled.	*enabled* indicates if this physical network interface is enabled. **Required: RULE:23.12.2.1-5** **Unsecure impact:** device dependent
unsecureMode	**Type:** xs:boolean **Card.:** Opt.		Read-only	*unsecureMode* is a read-only attribute that indicates that one or more configurations

	Default: *None*			in this XML do not meet the LXI minimum requirements for secure device operation. See the LXI Security Extended Function for the criteria to determine if a device is in unsecureMode. Several configurations within the API schemas are documented as placing the instrument into Unsecure Mode, however, the overall determination shall be done by the instrument per the requirement in the LXI Security Extended function. **Required: RULE:23.12.2.1-6** (read-only) **Unsecure impact:** NA
other Unsecure Protocols Enabled	**Type:** xs:boolean **Card.:** Opt. **Default:** true	No change, unless the protocol represented must be enabled to re-establish ethernet communication.		*otherUnsecureProtocolsEnabled* represents the state of various device-specific protocols that are beyond the scope of LXI. As a group, controllable unsecure protocols beyond the scope of the LCI Common Configuration including: 1) LXI specified instrument Common Configuration API, 2) Device specific extensions to the instrument Common Configuration are reflected by the state of this attribute. If *otherUnsecureProtocolsEnabled* is true, various device unsecure protocols beyond the scope of the LXI Common Configuration are permitted to be enabled. If *otherUnsecureProtocolsEnabled* is false, all controllable unsecure protocols not enabled by the LXI Common Configuration. For the purpose of *otherUnsecureProtocolsEnabled*, a secure protocol is a protocol that authenticates the server and encrypts data. Client authentication is not required. **RULE:23.12.2.1-7** LXI Secure devices shall document the protocols that are controlled by this attribute. **RULE:23.12.2.1-8** If the device does not implement any other unsecure protocols, then on a GET, *otherUnsecureProtocolsEnabled* shall return false. However, if written true, such a device shall either fail the PUT or indicate unsecure mode is True. **Required: RULE:23.12.2.1-9**

				Unsecure impact: Unsecure when the attribute is True. Device determined when the attribute is false
Any Attribute	**Type:** **Card.:** **Default:**	*Any type* *Optional* *NA*	**RULE:23.12.2.1-10** Those settings necessary to re-establish ethernet communication with the instrument shall be enabled.	Arbitrarily typed elements may be included for devices to represent device-specific configuration. **RULE:23.12.2.1-11** The impact of these configurations on the device secure mode are determined by the device vendor. However, if unsecure protocols are enabled, the device shall indicate it is in unsecure mode. **Required:** No **Unsecure impact:**

23.12.2.2 **Sub-elements**

The following must occur in this order:

Element	Type	Cardinality	Requirements
Network	lxi:Network	Optional	**RULE:23.12.2.2-1** *Network* is required. Interfaces that are not LXI Conformant are required to implement this element, however, they are only required to implement the *Interface/@Enabled* attribute.
HTTP	lxi:HTTP	Optional unbounded	**RULE:23.12.2.2-2** *HTTP* is optional, however devices that implement HTTP are require to fully implement this element. **RULE:23.12.2.2-3** If multiple HTTP elements are present, each shall have a different port. Additional instances of this element may be used to provide independent control of multiple servers (although each must be on a different port). This may be useful, for instance, if separate servers are setup for the API and the human interface. **OBSERVATION:** *Multiple services can be enabled on a single port. by including multiple instances of the HTTP/Service element.* **OBSERVATION:** *Devices may place restrictions on which services may be enabled or disabled on various ports. For instance, some devices may require that all API services be enabled or disabled together. To do so, the HTTP/Service element for each service must be set the same.*
HTTPS	lxi:HTTPS	Optional unbounded	**RULE:23.12.2.2-4** *HTTPS* is required. **RULE:23.12.2.2-5** If multiple HTTPS elements are present, each shall have a different port. Additional instances of this element may be used to provide independent control of multiple servers (although each

			must be on a different port). This may be useful, for instance, if separate servers are setup for the API and the human interface. **OBSERVATION:** *Multiple services can be enabled on a single port. by including multiple instances of the HTTPS/Service element.* **OBSERVATION:** *Devices may place restrictions on which services may be enabled or disabled on various ports. For instance, some devices may require that all API services be enabled or disabled together. To do so, the HTTPS/Service element for each service must be set the same.*
SCPIRaw	lxi:SCPIRaw	Optional unbounded	**RULE:23.12.2.2-6** At least one instance of *SCPIRaw* shall be accepted by devices that implement a SCPIRaw Command and Control connection. A separate instance of *SCPIRaw* is used for each port at which a SCPIRaw server is running.
Telnet	lxi:Telnet	Optional unbounded	**RULE:23.12.2.2-7** At least one instance of *Telnet* shall be accepted by devices that implement the Telnet Command and Control connection. A separate instance of *Telnet* is used for each port at which a Telnet server is running. **OBSERVATION:** *This may be useful, for instance, if one server provides access to a SCPI parser, and another to the device operating system shell.*
SCPITLS	lxi:SCPITLS	Optional unbounded	**RULE:23.12.2.2-8** At least one instance of *SCPITLS* shall be accepted by devices that implement a SCPITLS Command and Control connection. A separate instance of *SCPITLS* is provided for each port at which a SCPITLS server is running.
HiSLIP	lxi:HiSLIP	Optional	**RULE:23.12.2.2-9** *HiSLIP* shall be accepted by devices that implement the LXI HiSLIP extended function. Only a single instance of *HiSLIP* is permitted because a single instance of the protocol supports an arbitrary number of instances of servers at an arbitrary number of subaddresses.
VXI11	lxi:VXI11	Optional	**RULE:23.12.2.2-10** *VXI11* shall be accepted by devices that implement a VXI-11 Command and Control connection. Only a single instance of the VXI-11 protocol can be created on an interface.
Any element	*Any type*	Optional unbounded	Arbitrary subelements may be included in the *Interface* element. This enables devices to represent configuration capabilities not included in this XML. **RULE:23.12.2.2-11** If a device receives a well-formed extension element it does not recognize, it shall ignore it. **RULE:23.12.2.2-12** On a GET, devices are permitted to express arbitrary configuration with extension elements,

however such a device shall accept configuration using those elements.

RULE:23.12.2.2-13 Any element that controls a protocol that impacts the unsecure mode shall include an *unsecureEnabled* attribute. Setting this false shall disable the protocol or disable the unsecure behavior. The device shall report UnsecureMode true, when any protocol has *unsecureEnabled* true.

OBSERVATION: *It is possible that other configurations in the extension protocol make it secure in reality, however, setting the unsecureEnabled attribute true shall make the device report UnsecureMode true.*

23.12.3 Network

Network contains various settings associated with the Ethernet interface.

The settings in *Network* generally may be common to all instruments in a system. Settings that are generally device specific or are automatically configured such as the device Ethernet address are in the LXI Device Specific Configuration.

See the LXI Device specification for details about the management of various Ethernet settings.

The *Network* complex type has **no attributes**

23.12.3.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
IPv4	lxi:IPv4	Optional	**RULE:23.12.3.1-1** *IPv4* is required.
IPv6	lxi:IPv6	Optional	**RULE:23.12.3.1-2** *IPv6* is required by devices that implement the IPv6 Extended Function.

23.12.4 IPv4

IPv4 represents the state of the IP version 4 capabilities of the device.

23.12.4.1 Attributes

Attribute	Syntax		LCI	Description
enabled	**Type:**	xs:boolean	True	*enabled* generally enables or disables IPv4 operation.
	Card.:	Opt.		**Required: RULE:23.12.4.1-1**
	Default:	true		**Unsecure impact:** Does not impact unsecure mode
autoIPEnabled	**Type:**	xs:boolean	True	*autoIPEnabled* represents the state of the Link Local Addressing capability in the device. If enabled, the device may acquire

	Card.: Opt. **Default:** *None*		an address using Link Local Address. Link Local addresses supersede static values configured in the LXI Device Specific Configuration. **RULE:23.12.4.1-2** If omitted, and *DHCPEnabled* is present, the device uses the same state as *DHCPEnabled*. **OBSERVATION:** *If a device has no static IP address configured and both autoIPEnabled and DHCPEnabled are disabled, the device could be no longer reachable via IPv4 on this interface. Devices may either permit this case, generate an error and reject the schema, or leave the configuration pending till a static address is provided.* **OBSERVATION:** *In some implementations autoIPEnabled and DHCPEnabled are coupled, that is, both must be enabled or disabled.* **Required: RULE:23.12.4.1-3** **Unsecure impact:** Does not impact unsecure mode
DHCPEnabled	**Type:** xs:boolean **Card.:** Opt. **Default:** *None*	True	*DHCPEnabled* represents the state of the device DHCP protocol. If True, configuration is accepted via the DHCP protocol. If DHCP is enabled, the device will accept IPv4 configuration from a DHCP server. DHCP configuration supersedes static values configured in the LXI Device Specific Configuration. **RULE:23.12.4.1-4** If omitted, and *autoIPEnabled* is present, the device uses the same state as *autoIPEnabled*. **OBSERVATION:** *If a device has no static IP address configured and both autoIPEnabled and DHCPEnabled are disabled, the device could be no longer reachable via IPv4 on this interface. Devices may either permit this case, generate an error and reject the schema, or leave the configuration pending till a static address is provided.* **OBSERVATION:** *In some implementations autoIPEnabled and DHCPEnabled are coupled, that is, both must be enabled or disabled.* **Required: RULE:23.12.4.1-5**

				Unsecure impact: Does not impact unsecure mode
mDNSEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*mDNSEnabled* represents the state of the multicast DNS responder in the device. The multicast DNS responder permits the device to be discovered and identified by clients. In some implementations the mDNS capability is coupled between IPv4 and IPv6. For those devices, the configuration of the *IPv4/@mDNSEnabled* and the *IPv6/@mDNSEnabled* must be the same. If *mDNSEnabled* is absent, and *IPv6/@mDNSEnabled* is present, then *mDNSEnabled* takes on the same value as *IPv6/@mDNSEnabled*. **Required: RULE:23.12.4.1-6** **Unsecure impact:** Does not impact unsecure mode
dynamic DNSEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*dynamicDNSEnabled* represents the state of the dynamic DNS capability. Dynamic DNS is used to publish the hostname of the device to DNS. If *dynamicDNSEnabled* is absent, and *IPv6/@dynamicDNSEnabled* is present, then *dynamicDNSEnabled* takes on the same value as *IPv6/@dynamicDNSEnabled*. **RULE:23.12.4.1-7** Dynamic DNS is optional for LXI devices. Therefore, if not implemented, the device shall ignore this attribute on a PUT. **RULE:23.12.4.1-8** Devices that do not implement dynamic DNS shall omit this attribute on a GET. **RULE:23.12.4.1-9** The *dynamicDNSEnabled* attribute shall be implemented irrespective of if IPv6 dynamic DNS is implemented. **Required: RULE:23.12.4.1-10** **Unsecure impact:** Does not impact unsecure mode
pingEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*pingEnabled* represents the state of the IPv4 ICMP ping responder. In some IPv6 implementation the ICMPv4 ping capability is coupled to the ICMPv6 ping. For those devices, the configuration

				of the *IPv4/@PingEnable* and the *IPv6/@PingEnabled* must be the same. If *pingEnabled* is absent, and *IPv6/@pingEnabled* is present, then *pingEnabled* takes on the same value as *IPv6/@pingEnabled*. **Required: RULE:23.12.4.1-11** **Unsecure impact:** Does not impact unsecure mode
Any Attribute	**Type:** *Any type* **Card.:** *Optional* **Default:** *NA*		No change unless the configured attribute is necessary to re-establish ethernet communication.	Arbitrary extension attributes may be included to provide device-specific IPv4 configuration that is beyond the scope of the LXI requirements. **RULE:23.12.4.1-12** LXI devices shall ignore extension attributes they do not recognize. **Required:** No **Unsecure impact:** Vendor determined

The IPv4 complex type has **no subelements**

23.12.5 IPv6

IPv6 represents the state of the IP version 6 capabilities of the device.

RULE:23.12.5-1 Since *IPv6* is required in devices that implement the LXI IP Version 6 Extended Function, the required attributes are only required in implementations that implement *IPv6.*

RULE:23.12.5-2 Devices shall implement *IPv6/@enabled*. If the device does not implement IPv6 it shall always return false. If *LXICommonConfiguration/@strict* attribute is false such a device ignores the *IPv6* element on a PUT.

23.12.5.1 Attributes

Attribute	Syntax		LCI	Description
enabled	**Type:**	xs:boolean	True	*enabled* generally enables or disables IPv6 capability. **Required: RULE:23.12.5.1-1** **Unsecure impact:** Does not impact unsecure mode
	Card.:	Opt.		
	Default:	true		
DHCPEnabled	**Type:**	xs:boolean	True	*DHCPEnabled* represents the state of the device IPv6 DHCP protocol. If True, configuration is accepted via the DHCP protocol. If DHCP is enabled, the device will accept IPv6 configuration from a DHCP server, which supercedes static values configured in the LXI Device Specific Configuration.
	Card.:	Opt.		
	Default:	true		

				See the note on IPv6 autoconfiguration under *Network/IPv6*. **Required: RULE:23.12.5.1-2** **Unsecure impact:** Does not impact unsecure mode
RAEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*RAEnabled* represents the state of address generation based on the router advertisement. See the note on IPv6 autoconfiguration under *Network/IPv6*. **Required: RULE:23.12.5.1-3** **Unsecure impact:** See @privacyModeEnabled.
static Address Enabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*staticAddressEnabled* indicates if the device uses the static address configured with *LXIDeviceSpecificConfiguration/IPv6/StaticAddress*. **OBSERVATION:** *There might not be a static address configured.* **Required: RULE:23.12.5.1-4** **Unsecure impact:** Does not impact unsecure mode.
privacy Mode Enabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*privacyModeEnabled* indicates if the MAC address is included in the IPv6 address generation. When *privacyModeEnabled* is enabled, neither the link local address, unique local address nor the SLAAC-generated addresses include the device MAC address. See the *Observation* on IPv6 autoconfiguration under *Network/IPv6*. **Required: RULE:23.12.5.1-5** **Unsecure impact:** If False the device is in unsecure mode.
mDNSEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*mDNSEnabled* represents the state of the IPv6 multicast DNS responder in the device. The multicast DNS responder permits the device to be discovered and identified by clients. In some implementations the mDNS capability is coupled between IPv4 and IPv6. For those devices, the configuration of the *IPv4/@mDNSEnabled* and the *IPv6/@mDNSEnabled* must be the same. If *mDNSEnabled* is absent, and *IPv4/@mDNSEnabled* is present, then *mDNSEnabled* takes on the same value as *IPv4/@mDNSEnabled*.

				Required: RULE:23.12.5.1-6 **Unsecure impact:** Does not impact unsecure mode
dynamic DNSEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*dynamicDNSEnabled* represents the state of the IPv6 dynamic DNS capability used to publish the hostname of the device. If *dynamicDNSEnabled* is absent, and *IPv4/@dynamicDNSEnabled* is present, then *dynamicDNSEnabled* takes on the same value as *IPv4/@dynamicDNSEnabled*. **RULE:23.12.5.1-7** Dynamic DNS is optional for LXI devices. Therefore, if not implemented, the device shall ignore this attribute on a PUT. **RULE:23.12.5.1-8** Devices that do not implement dynamicDNS shall omit this attribute on a GET. **Required: RULE:23.12.5.1-9** Attribute shall be implemented irrespective of if IPv4 dynamic DNS is implemented. **Unsecure impact:** Does not impact unsecure mode
pingEnabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	True	*pingEnabled* represents the state of the IPv6 ICMP ping function. If *pingEnabled* is absent, and *IPv4/@pingEnabled* is present, then *pingEnabled* takes on the same value as *IPv4/@pingEnabled*. **OBSERVATION:** *The common IPv4 practice of blocking ICMP packets as a supposed security measure is not recommended on IPv6, as IPv6 functioning depends on ICMPv6 for error messages, path MTU discovery, multicast group management and Neighbor Discovery. IPv6 also relies upon multicast availability, which impacts firewalls, intrusion detection and access control rules.* **OBSERVATION:** *On some devices IPv6/@pingEnabled must match the IPv4/@pingEnabled state*. **Required: RULE:23.12.5.1-10** **Unsecure impact:** Does not impact unsecure mode
Any Attribute	**Type:** **Card.:** **Default:**	*Any type* *Optional* *NA*	No change unless changing the configured attribute is necessary to re-establish ethernet	Arbitrary extension attributes may be included to provide device specific IPv6 configuration that is beyond the scope of the LXI requirements. **RULE:23.12.5.1-11** LXI devices shall ignore extension attributes they do not recognize. **Required:** No

communication **Unsecure impact:** Vendor determined
.

The IPv6 complex type has **no subelements**

23.12.6 HTTP

HTTP represents the configuration of the unsecure HTTP server including general behavior and the services available on the server.

Additional instances of the *HTTP* element are used to configure additional servers on other ports, however, a single *HTTP* element configures all servers on a given port.

Some endpoints may be used by multiple services. If so, those endpoints are enabled if any service requiring them is enabled.

RULE:23.12.6-1 If no services are specified the server at this port is disabled.

RULE:23.12.6-2 If any service is enabled that permits changing the device configuration over an unencrypted connection the device is in unsecure mode.

OBSERVATION: *Since the normal behavior of HTTP is to forward secure URLS to HTTPS, it is not common for enabling HTTP to put the device into unsecure mode.*

RULE:23.12.6-3 Devices that implement the optional unsecure HTTP interface shall not change the HTTP/@operation state nor the enabled services on LCI unless necessary to re-establish communication.

23.12.6.1 Attributes

Attribute	Syntax	LCI	Description
operation	**Type:** restriction of: xs:string **Card.:** Opt. **Default:** enable	No change unless changing the configured attribute is necessary to re-establish ethernet communication. For instance, to enable the LXI API.	*operation* controls if the HTTP server is enabled, disabled, or if it forwards all requests to HTTPS. **enable** Enables the HTTP server, although secure pages shall redirect to HTTPS. **disable** Disables the HTTP server irrespective of the enabled services. No forwarding function is active. **redirectAll** All accesses to the HTTP server are redirected to HTTPS. **Required: RULE:23.12.6.1-1** Devices that implement the unsecure HTTP protocol shall implement at least the disable and redirectAll settings of @operation.

				Unsecure impact: True if access to instrument configuration is enabled via any HTTP service.
port	**Type:** **Card.:** **Default:**	xs:int Opt. 80	**RULE:23.12.6.1-2** The LCI HTTP port for the LXI Web interface and the LXI API services shall be 80.	TCP port of the HTTP server. **Required: RULE:23.12.6.1-3** **Unsecure impact:** Does not impact unsecure mode.

23.12.6.2 **Sub-elements**

The following must occur in this order:

Element	Type	Cardinality	Requirements
Service	lxi:Service	Optional unbounded	Each *Service* element indicates the state of the HTTP service indicated by the *Service/@name* attribute. Only those services with the *Service/@enabled* attribute set to *true* are enabled on this HTTP server. *Service* elements with the *Service/@enabled* attribute set to false indicate the service is disabled. **OBSERVATION:** *Users should be cautious emnabling authentication over HTTP since it may expose unencrypted credentials.* **OBSERVATION:** *Any service not explicitly enabled is disabled.* **RULE:23.12.6.2-1** When the device is queried, it shall provide a *Service* element for each service provided by the device, with the *Service/@enable* attribute indicating those that are currently enabled.

23.12.7 **HTTPS**

HTTPS configures the secure HTTPS server. That is, the HTTP server that serves content using TLS.

Some endpoints may be used by multiple services. If so, they are enabled when any service that requires the endpoint is enabled.

Disabled elements are used on a read to indicate the schemes implemented by the device.

Disabled elements on a write explicitly indicate that the corresponding scheme is disabled. However, omitting the element indicating the schema has the same affect.

If a device is configured to require application-level authentication it may report the connection is not unsecure.

RULE:23.12.7-1 The HTTPS web human interface content served by LXI Secure devices shall be a superset of the content available via HTTP. That is, a device is not permitted to only offer a subset of the HTTP human interface over the secure HTTPS connection.

RULE:23.12.7-2 If no services are enabled, then the HTTPS server is disabled.

In addition to the LXI-required HTTP client authentication, LXI devices should provide application-level authentication.

RULE:23.12.7-3 If the device is using application-level client authentication, none of the subelements indicating HTTP client authentication need to be enabled in the *HTTPS* element.

RULE:23.12.7-4 When returning the LXI Common Configuration, if a scheme is implemented, then the element representing that scheme shall be present. This permits clients to determine what schemes are available on the device.

RULE:23.12.7-5 After an LCI, the security scheme is not changed.

RULE:23.12.7-6 On LCI the LXI Web interface and the LXI API services shall be enabled.

23.12.7.1 **Attributes**

Attribute	Syntax	LCI	Description
port	**Type:** xs:int **Card.:** Opt. **Default:** 443	**RULE:23.12.7.1-1** The default HTTPS port shall be 443 for the Human Interface and the LXI API Service.	TCP port of the HTTPS server. **Required: RULE:23.12.7.1-2** **Unsecure impact:** Does not impact unsecure mode
client Authentication Required	**Type:** xs:boolean **Card.:** Opt. **Default:** false	No change	*clientAuthenticationRequired* indicates if clients are required to authenticate as configured in this element. *clientAuthenticationRequired* does not impact the API-LXISecurity service which always requires client authentication. Note that client's presenting the API Key are regarded as authentic. **OBSERVATION:** *If the service is using application level authentication, this attribute may be the only indication in the schema that the HTTPS server communication is secure.* **Required: RULE:23.12.7.1-3** **Unsecure impact:** Does not impact unsecure mode.

23.12.7.2 **Sub-elements**

The following must occur in this order:

Element	Type	Cardinality	Requirements
Service	lxi:Service	Optional unbounded	A *Service* element with the *Service/@enabled* attribute set to true is included for each service enabled on this HTTP(S) server. **RULE:23.12.7.2-1** When the device is queried, it shall provide a *Service* element for each service provided by the device, with the *Service/@enable* attribute indicating those that are currently enabled.

23.12.8 Service

The *Service* element is used with the *HTTP* and *HTTPS* elements to indicate the services available on a device and if they are currently enabled.

23.12.8.1 Attributes

Attribute	Syntax		LCI	Description
name	**Type:**	xs:string	NA	*name* indicates the name of the service.
	Card.:	Req.		**RULE:23.12.8.1-1** LXI Service names are case sensitive. LXI Security specifies the following services:
	Default:	NA		Human-Interface: Indicates the endpoints required to serve a human interface to a browser Required of all LXI Security Devices.
				API-LXISecurity: Indicates the LXI API Extended function endpoints required by the LXI Security extended function. Required of all LXI Security Devices.
				API-Device: Indicates the endpoints used to implement various device-specific APIs. *API-Device* is required of all LXI Security devices that provide a device-specific API on this protocol. More fine-grained device-specific control is permitted as well.
				other: Devices may define additional services that provide more granular control of enabled services or specify additional services. *other* service declarations are optional. **OBSERVATION:** *clients can discover the available services by reading back the LXI Common Configuration. Although documentation of the device behavior is in product-specific documents.* **OBSERVATION:** *where servers define granular services that are a subset of other services, the presence of the less granular service enables all of the subset services.*
				Required: RULE:23.12.8.1-2 **Unsecure impact:** None
enabled	**Type:**	xs:boolean	NA	*enabled* indicates if the designated service is enabled.
	Card.:	Req.		**Required: RULE:23.12.8.1-3** Note this attribute is syntactically required.
	Default:	NA		**Unsecure impact:** Device determined

Any Attribute	**Type:** **Card.:** **Default:**	*Any type* *Optional* *NA*	NA	Devices may include attributes to further configure the service. **RULE:23.12.8.1-4** Devices that do not understand additional attributes shall ignore them. **Required:** No **Unsecure impact:** Device determined

23.12.8.2 **Sub-elements**

The following must occur in this order:

Element	Type	Cardinality	Requirements
Basic	lxi:AuthenticationMechanism	Optional	*Basic* indicates HTTPS Basic authentication per RFC7617 is enabled. **RULE:23.12.8.2-1** Devices shall implement *Basic*. When *Basic* is configured, devices are permitted to not be in unsecure mode.
Digest	lxi:AuthenticationMechanism	Optional	*Digest* indicates Digest authentication per RFC7616 is enabled. **RULE:23.12.8.2-2** Devices are permitted to not implement *Digest*, however this syntax shall be accepted and produce an error if turned on and not implemented. When *Digest* is configured, devices are permitted to not be in unsecure mode.
Any element	*Any type*	Optional unbounded	This element is provided to enable devices to extend the list of HTTP authentication schemes with additional elements to configure capabilities not included in the definition of the LXI Common Configuration. **OBSERVATION:** *The rules in the lxi:AuthenticationMechanism element section require the type of these extension elements to be either lxi:AuthenticationMechanism or an extension of lxi:AuthenticationMechanism.* **RULE:23.12.8.2-3** The default value of the *enabled* attribute of extension elements shall be *True* so that the presence of the element without a value indicates the mechanism is enabled. The element name should match the authentication scheme in the IANA HTTP Authentication Schemes Registry.

RULE:23.12.8.2-4 Any extension HTTPS client-authentication scheme is permitted with unsecure mode false.

23.12.9 SCPIRaw

SCPIRaw configures a single SCPIRaw connection. Additional instances of *SCPIRaw* may be used to configure additional SCPIRaw servers at different TCP ports.

SCPIRaw refers to a TCP port that accepts SCPI commands and queries without IEEE 488.2 meta-messages.

Devices are permitted to enable an arbitrary number of SCPIRaw ports, however, each must have a different port number and an additional *SCPIRaw* element to describe it.

RULE:23.12.9-1 When the device receives an LXI Common Configuration, only those SCPIRaw ports indicated and enabled shall be available on the device.

RULE:23.12.9-2 When the device reports its configuration, an instance of *SCPIRaw* shall be provided for each active SCPIRaw connection.

Devices should permit multiple clients to connect to a single SCPIRaw port.

RULE:23.12.9-3 *SCPIRaw* is required if the device implements SCPIRaw connections.

23.12.9.1 Attributes

Attribute	Syntax		LCI	Description
enabled	**Type:**	xs:boolean	No change	*enabled* enables the SCPIRaw server at this address. **Required: RULE:23.12.9.1-1** **Unsecure impact: RULE:23.12.9.1-2** The device is operating in unsecure mode if SCPIRaw is enabled.
	Card.:	Opt.		
	Default:	true		
port	**Type:**	xs:int	No change	*port* specifies the port of this SCPIRaw server. The IANA registered port of 5025 is preferred for SCPI traffic. If additional instances of SCPIRaw are enabled by default on the device, their ports are device-specific. **Required: RULE:23.12.9.1-3** **Unsecure impact:** Does not impact unsecure mode
	Card.:	Opt.		
	Default:	5025		
capability	**Type:**	xs:int	Read-only	*capability* is a read-only attribute that indicates the approximate number of SCPIRaw ports that the client may configure. **Required:** capability is required on a read. **Unsecure impact:** NA
	Card.:	Opt.		
	Default:	*None*		

The SCPIRaw complex type has **no subelements**

23.12.10 SCPITLS

SCPITLS describes a single secure raw SCPI connection over TLS. Additional instances of *SCPITLS* may be used to configure additional secure raw SCPI servers at different TCP (TLS) ports.

Devices are permitted to enable an arbitrary number of secure raw SCPI ports using *SCPITLS*, however, each must have a different port number.

Devices should permit multiple clients to connect to a single secure raw SCPI port.

RULE:23.12.10-1 When the device receives an LXI Common Configuration, only those secure raw SCPI ports indicated and enabled shall be available on the device.

RULE:23.12.10-2 When the device reports its configuration, an instance of *SCPITLS* shall be included for each configured secure raw SCPI connection. If none are enabled, a single disabled *SCPITLS* element shall be returned to indicate to the client that the capability is available.

RULE:23.12.10-3 *SCPITLS* is required by LXI Security if the device implements secure raw SCPI connections.

23.12.10.1 Attributes

Attribute	Syntax	LCI	Description
enabled	**Type:** xs:boolean **Card.:** Opt. **Default:** true	No change	*enabled* enables the secure raw SCPI server at this port. **Required: RULE:23.12.10.1-1** **Unsecure impact:** Does not impact unsecure mode
port	**Type:** xs:int **Card.:** Req. **Default:** NA	No change	*port* specifies the port of this secure raw SCPI server. **Required: RULE:23.12.10.1-2** **Unsecure impact:** Does not impact unsecure mode
client Authentication Required	**Type:** xs:boolean **Card.:** Opt. **Default:** false	No change	*clientAuthenticationRequired* indicates if client authentication is required. Secure raw SCPI connections use mutual TLS (mTLS) for client authentication. The client certificate is authenticated based on the *Interface/ClientAuthentication/ClientCertAuthentication* element which must be configured if an active secure raw SCPI connection requires client authentication. If false, the device accepts SCPITLS connections without client authentication, although mTLS connections may still be supported. **Required: RULE:23.12.10.1-3** If secure raw SCPI client authentication is implemented it shall use ClientAuthentication configuration. **Unsecure impact:** Does not impact unsecure mode
capability	**Type:** xs:int **Card.:** Opt. **Default:** *None*	Read-only	*capability* is a read-only attribute. It indicates the approximate number of SCPITLS ports that the client may configure. **Required: RULE:23.12.10.1-4**

Unsecure impact: NA

The SCPITLS complex type has **no subelements**

23.12.11 Telnet

Telnet indicates the telnet connection. Telnet is commonly used for either Command and Control traffic or an operating system shell.

23.12.11.1 Attributes

Attribute	Syntax	LCI	Description
enabled	**Type:** xs:boolean **Card.:** Opt. **Default:** true	No change	*enabled* indicates if the Telnet server is enabled. **Required: RULE:23.12.11.1-1** **Unsecure impact:** If Telnet is enabled without requiring TLS (Telnet/@TLSRequired) the device is in unsecure mode.
port	**Type:** xs:int **Card.:** Opt. **Default:** 5024	No change	*port* specifies the Telnet server port. For Command and Control traffic, the IANA assigned port of 5024 should be used. **Required: RULE:23.12.11.1-2** **Unsecure impact:** Does not impact unsecure mode
TLSRequired	**Type:** xs:boolean **Card.:** Opt. **Default:** false	No change	*TLSRequired* indicates that telnet requires a secure TLS connection instead of TCP. **OBSERVATION:** *TLS only guarantees server (device) authentication. To require client authentication, @clientAuthenticationRequired must be true as well.* **RULE:23.12.11.1-3** If the device implements TLS on Telnet it shall include the *TLSRequired* attribute in the query response regardless of the state of *Telnet/@enabled*. **Required: RULE:23.12.11.1-4** TLSRequired shall be implemented if the device Telnet implementation supports TLS. **Unsecure impact:** The device is in unsecure mode unless enabled.
client Authentication Required	**Type:** xs:boolean **Card.:** Opt. **Default:** false	No change	*clientAuthenticationRequired* indicates that telnet exclusively uses mTLS. **OBSERVATION:** *If clientAuthenticationRequired is enabled, Telnet/@TLSRequired must be enabled as well. The unsecure Telnet USER and PASS are not used.* **RULE:23.12.11.1-5** The mTLS client certificate authentication configured in

				Interface/ClientAuthentication/ClientCertAuthentication shall be used. **RULE:23.12.11.1-6** If the device implements mTLS (client authentication) on telnet it shall include the *clientAuthenticationRequired* attribute in the query response regardless of the state of *Telnet/@enabled*. **Required: RULE:23.12.11.1-7** clientAuthenticationRequired shall be implemented if the device Telnet implementation supports TLS. **Unsecure impact:** Does not impact unsecure mode
capability	**Type:** **Card.:** **Default:**	xs:int Opt. *None*	Read-only	*capability* is a read-only attribute. It indicates the approximate number of Telnet ports that the client may configure. **Required: RULE:23.12.11.1-8** **Unsecure impact:** NA
Any Attribute	**Type:** **Card.:** **Default:**	*Any type* *Optional* *NA*	NA	Devices may further describe the telnet port, perhaps indicating if this server is SCPI or a command shell. **RULE:23.12.11.1-9** Devices that do not understand additional attributes shall ignore them. **Required:** No **Unsecure impact:** Device determined

The Telnet complex type has **no subelements**

23.12.12 HiSLIP

HiSLIP contains the configuration of the HiSLIP protocol. HiSLIP supports multiple servers on a port, each at a different subaddress. Therefore, this element contains the configuration of the only device HiSLIP port.

All HiSLIP servers, regardless of their subaddress use the configuration in this element.

23.12.12.1 Attributes

Attribute	Syntax		LCI	Description
enabled	**Type:** **Card.:** **Default:**	xs:boolean Opt. true	No change	*enabled* indicates if the HiSLIP server is enabled. **OBSERVATION:** *Disabling this server disables all the HiSLIP servers at every HiSLIP subaddress since they are all served from this port at the various subaddresses.* **Required: RULE:23.12.12.1-1** **Unsecure impact: RULE:23.12.12.1-2** The device is in unsecure mode unless both HiSLIP/@mustStartEncrypted and HiSLIP/@encryptionMandatory are true.
port	**Type:**	xs:int	No change	*port* indicates the TCP port from which the HiSLIP server is served.

	Card.:	Opt.		**Required: RULE:23.12.12.1-3** **Unsecure impact:** Does not impact unsecure mode
	Default:	4880		
must Start Encrypted	**Type:**	xs:boolean	No change	mustStartEncrypted controls the initial encryption. If enabled, a secure connection must be initially made to this server. It can be subsequently stepped down to an unsecure connection if encryptionMandatory is not true. It is erroneous to have *mustStartEncrypted* False and *HiSLIP/@encryptionMandatory* True. **Required: RULE:23.12.12.1-4** **Unsecure impact: RULE:23.12.12.1-5** The device is in unsecure mode if mustStartEncrypted is false.
	Card.:	Opt.		
	Default:	false		
encryption Mandatory	**Type:**	xs:boolean	No change	*encryptionMandatory* indicates that this HiSLIP Server must always have encryption on. That is, the connection must be started securely, and the encryption may not be subsequently turned off. It is erroneous to have *encryptionMandatory* True and *HiSLIP/@mustStartEncrypted* False. **Required: RULE:23.12.12.1-6** **Unsecure impact: RULE:23.12.12.1-7** The device is in unsecure mode if encryptionMandatory is false for any enabled HiSLIP servers.
	Card.:	Opt.		
	Default:	false		

23.12.12.2 **Sub-elements**

The following must occur in this order:

Element	Type	Cardinality	Requirements
Client Authentication Mechanisms	lxi:ClientAuthenticationMechanisms	Optional	**RULE:23.12.12.2-1** Devices that support the LXI Security Extended Function and the LXI HiSLIP Extended function shall support Client Authentication. A device may optionally provide client authentication using a higher protocol layer (for example, SCPI) to provide authentication when using ANONYMOUS.

23.12.13 ClientAuthenticationMechanisms

ClientAuthenticationMechanisms identifies the SASL mechanisms that are enabled for secure HiSLIP connections. The default of the *enabled* attribute for each element is true, therefore, its presence with no attributes enables the mechanism. The absence of an element disables the corresponding mechanism.

OBSERVATION: *ClientAuthenticationMechanisms does not affect the behavior of unsecure HiSLIP connections which may be enabled using HiSLIP/@mandatoryEncryption and HiSLIP/@mustStartEncrypted*.

OBSERVATION: *Client credentials are shared amongst the mechanisms and are described in the root ClientAuthentication element.*

RULE:23.12.13-1 the device shall include in its response each element that it implements, indicating a false enable attribute where disabled. Devices shall omit the elements that represent mechanisms they do not support.

RULE:23.12.13-2 Devices that implement device-specific SASL mechanisms shall follow the pattern of defining additional elements that enable and configure those mechanisms using the *AuthenticationMechanism* complex type, or types derived from it.

The *ClientAuthenticationMechanisms* complex type has **no attributes**

23.12.13.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
ANONYMOUS	lxi:AuthenticationMechanism	Optional	*ANONYMOUS* indicates that clients can authenticate using the SASL anonymous mechanism. **RULE:23.12.13.1-1** Devices that support LXI Security and the LXI HiSLIP Extended function shall support *ANONYMOUS*. **OBSERVATION:** *A device can optionally provide client authentication using a higher protocol layer (e.g., SCPI) when using ANONYMOUS.* Configuring *ANONYMOUS* does not put the device into unsecure mode. **RULE:23.12.13.1-2** The IVI-6.5 SASL Mechanism Specification details specific requirements for SASL mechanisms. Devices shall comply with the IVI device requirements.
PLAIN	lxi:AuthenticationMechanism	Optional	*PLAIN* indicates that clients can authenticate using the SASL PLAIN mechanism. **RULE:23.12.13.1-3** The IVI-6.5 SASL Mechanism Specification details the specific device and client requirements for the generation of usernames and passwords. Devices shall comply with the IVI device requirements. **RULE:23.12.13.1-4** Devices that support LXI Security and the LXI HiSLIP Extended function shall support *PLAIN*. Configuring *PLAIN* does not put the device into unsecure mode.
SCRAM	lxi:AuthenticationMechanism	Optional	*SCRAM* indicates that clients can authenticate using the SASL SCRAM (Salted

			Challenge Response Authentication Mechanism) mechanism. Two attributes that are used to configure the SCRAM mechanism are located on the element LXICommonConfiguration/ClientAuthentication. See them for additional details. **RULE:23.12.13.1-5** The IVI 6.5 SASL Mechanism Specification details the specific device and client requirements for the use of the SASL SCRAM mechanism with HiSLIP. Devices shall comply with the IVI device requirements. **RULE:23.12.13.1-6** Devices that support LXI Security and the LXI HiSLIP Extended function shall support *SCRAM*. Configuring *SCRAM* does not put the device into unsecure mode.
MTLS	lxi:AuthenticationMechanism	Optional	*MTLS* indicates that devices authenticates TLS clients using TLS mutual authentication (mTLS). **OBSERVATION:** *mTLS connections provide client authentication outside of the SASL mechanisms, therefore SASL refers to mTLS as an EXTERNAL mechanism.* Configuring *MTLS* does not put the device into unsecure mode.
Any element	*Any type*	Optional unbounded	Other extension elements may be included to configure authentication mechanisms that are beyond the scope of the LXI specification. Where registered SASL mechanisms are used, the IANA designation for those mechanisms should be used in the XML. **RULE:23.12.13.1-7** Devices shall ignore mechanisms that they do not implement. Devices that implement extension mechanisms per this attribute shall include them in the response.

23.12.14 AuthenticationMechanism

AuthenticationMechanism specifies a type of client authentication. It is used for both HiSLIP SASL mechanisms and HTTPS security schemes.

AuthenticationMechanism/@enabled indicates if the mechanism is currently enabled. The tag for the element indicates the specific mechanism or scheme.

HTTP client authentication is described in RFC7235. IANA maintains a list of HTTP authentication schemes, the IANA names of those schemes are generally used as the tag name of the element used to enable the HTTP authentication scheme.

SASL mechanisms are generally specified using the registered SASL mechanism names. For instance, the *PLAIN* SASL mechanism is controlled with an element with the tag (name) *PLAIN*, and the type *AuthenticationMechanism*. The *PLAIN* mechanism is enabled if *PLAIN/@enabled* attribute is *true*.

RULE:23.12.14-1 Where possible, additional client authentication capabilities beyond the scope of the LXI Security Extended Function shall be created using this type. However, if those capabilities require additional configuration, they shall define their own type by extending the *AuthenticationMechanism* ComplexType.

23.12.14.1 Attributes

Attribute	Syntax	LCI	Description
enabled	**Type:** xs:boolean **Card.:** Opt. **Default:** true	No change	*enabled* indicates that the SASL mechanism or HTTP scheme is enabled. **RULE:23.12.14.1-1** On LCI, the enabled mechanisms do not change. **Required: RULE:23.12.14.1-2** **Unsecure impact:** NA
Any Attribute	**Type:** *Any type* **Card.:** *Optional* **Default:** *NA*	See the usage of the defined mechanism.	Additional attributes that define SASL mechanisms or HTTPS schemas beyond the scope of LXI may include additional attributes to define them. **Required:** Not required. **Unsecure impact:**

The AuthenticationMechanism complex type has **no subelements**

23.12.15 VXI11

VXI11 configures the VXI-11 protocol.

23.12.15.1 Attributes

Attribute	Syntax	LCI	Description
enabled	**Type:** xs:boolean **Card.:** Opt. **Default:** true	No change	*enabled* state of the VXI11 server at this address. **Required:** Not required. **Unsecure impact: RULE:23.12.15.1-1** The device is in unsecure mode if VXI-11 is enabled.

The VXI11 complex type has **no subelements**

23.12.16 ClientAuthentication

ClientAuthentication contains client authentication information. That is, information used by the device to determine if the identity proffered by clients attempting to connect to it is authentic.

RULE:23.12.16-1 Information in *ClientAuthentication* shall be used by all protocols that provide client authentication. For instance, a certificate thumbprint that the device accepts for HiSLIP EXTERNAL authentication, will also be accepted for telnet mTLS.

OBSERVATION: *Devices may require that all ClientCredentials are re-sent when the @scramHashCount is changed. Because of this requirement, although this attribute is most closely associated with LXICommonConfiguration/HiSLIP/ClientAuthentication/SCRAM, it is located here so that changes to the @scramHashCount are directly associated with the credentials that must be hashed.*

In addition, @scramChannelBindingRequired is located on this element to retain its association with the @scramHashIterationCount.

OBSERVATION: *Devices may also have mechanisms beyond the scope of the LxiCommonConfiguration to manage the passwords.*

23.12.16.1 Attributes

Attribute	Syntax		LCI	Description
scram Hash Iteration Count	**Type:** **Card.:** **Default:**	xs:int Opt. *None*	No change	*scramHashIterationCount* sets the minimum iteration count that SCRAM uses to hash the client credentials. The default value of this is device dependent, but should be chosen sufficiciently high that clients cannot successfully perform brute force attacks. At the time of this writing RFC 7677 recommends a minimum of 4096 for SHA-256, although much larger values are reasonable for LXI devices. **OBSERVATION:** *Devices are permitted to use a higher value for scramHashIterationCount. The actual iteration count used by the device is indicated in the SCRAM protocol.* **Required: RULE:23.12.16.1-1** Required for devices that support the SCRAM SASL mechanism via the LXICommonConfiguration. **Unsecure impact:** NA
scram Channel Binding Required	**Type:** **Card.:** **Default:**	xs:boolean Opt. *None*	No change	*scramChannelBindingRequired* specifies if the device permits the client to connect with a non-channel-bound version of SCRAM. For instance, for a device that supports SCRAM with SHA-256 hashes: if false, then SCRAM-SHA-256 would be accepted in addition to SCRAM-SHA-256-PLUS. If true, only SCRAM-SHA-256-PLUS would be accepted by the device. **Required: RULE:23.12.16.1-2** Required for devices that support the SCRAM SASL mechanism via the LXICommonConfiguration. **Unsecure impact:** NA

23.12.16.2 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
Client Credential	lxi:ClientCredential	Optional unbounded	**RULE:23.12.16.2-1** Required.
Client Cert Authentication	lxi:ClientCertAuthentication	Optional	**RULE:23.12.16.2-2** Required.
Any element	*Any type*	Optional unbounded	Extension elements may be included to enable devices to specify types of ClientAuthentication beyond the scope of LXI.

23.12.17 ClientCredential

ClientCredential contains an individual user with an optional password and an indication if this used has API Access rights.

@password and *@APIAccess* are optional since they are write-only fields and are not included in device responses for reasons of secrecy.

23.12.17.1 Attributes

Attribute	Syntax		LCI	Description
user	**Type:**	xs:string	No change	*user* that may be authenticated on the device.
	Card.:	Req.		**RULE:23.12.17.1-1** LXI devices shall accept *user* names composed of alpha-numeric strings. User names shall be case-sensitive.
	Default:	NA		**RULE:23.12.17.1-2** The IVI-6.5 SASL Mechanism Specification details the specific device and client requirements for the generation of usernames and passwords. Devices shall comply with the IVI device requirements.
				Required: RULE:23.12.17.1-3
				Unsecure impact: NA
password	**Type:**	xs:string	No change	*password* contains the password associated with this user name.
	Card.:	Opt.		*password* is a write-only attribute. On a write, the absence of a password indicates that no change is to be made to the devices stored password. If the user has not already defined a password and the password is absent, then the user is not authenticated until a password is configured. The password may be configured using this mechanism or other device-specific mechanisms beyond the scope of LXI Security.
	Default:	*None*		**RULE:23.12.17.1-4** The IVI-6.5 SASL Mechanism Specification details the specific device and client

				requirements for the generation of usernames and passwords. Devices shall comply with the IVI device requirements. **Required: RULE:23.12.17.1-5** **Unsecure impact:** NA
APIAccess	**Type:** **Card.:** **Default:**	xs:boolean Opt. false	No change	*APIAccess* indicates if this user is authorized to use the API. If *true*, this user credential permits the client to use the API. If *APIAccess* is *false*, this credential is not sufficient to permit the client to use the API. On a write, the absence *APIAccess* indicates that no change is to be made to the users stored *APIAccess* value. **Required: RULE:23.12.17.1-6** **Unsecure impact:** NA

The ClientCredential complex type has **no subelements**

23.12.18 ClientCertAuthentication

Configures client certificate authentication.

RULE:23.12.18-1 Devices shall accept client certificates as valid if they are signed by a root certificate specified in this element, or if they have a thumbprint that matches a thumbprint specified in this element.

The *ClientCertAuthentication* complex type has **no attributes**

23.12.18.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
RootCertPEM	xs:string	Optional unbounded	*RootCertPEM* has a single root certificate the device shall use to validate client certificates. Any client certificate that is signed by a trust authority described in one of these root certificates shall be treated as authentic by the device. Certificates are in PEM format, represented in XML as strings. PEM format is a Base64 ASCII encoding of the binary certificate. PEM Format is described in RFC 7468. **RULE:23.12.18.1-1** Root certification PEMs shall be semantically validated. For instance, expired root certificates shall not be used. **RULE:23.12.18.1-2** *RootCertPEM* shall be supported.
Cert Thumbprint	lxi:CertThumbprint	Optional unbounded	Each instance of this element has the thumbprint of a client certificate. Client certificates with this

thumbprint shall be treated as authentic by the device. Authenticated certificates still require semantic validation, for instance, expired certificates shall not be used.

The thumbprint is a hash of the full binary device certificate. The hash function is specified in the CertThumbprint element.

RULE:23.12.18.1-3 *CertThumbprint* shall be supported.

23.12.19 CertThumbprint

CertThumbprint contains a certificate thumbprint. A certificate thumbprint is a hash of a DER encoded X.509 certificate that is used to recognize a specific certificate.

23.12.19.1 Attributes

Attribute	Syntax		LCI	Description
hash	**Type:**	xs:string	No change	*hash* indicates the hash function used to create this thumbPrint.
	Card.:	Opt.		**Required: RULE:23.12.19.1-1**
	Default:	SHA-256		**Unsecure impact:** NA
thumbPrint	**Type:**	xs:base64Binary	No change	*thumbPrint* contains the certificate thumbPrint.
	Card.:	Req.		**Required: RULE:23.12.19.1-2**
	Default:	NA		**Unsecure impact:** NA

The CertThumbprint complex type has **no subelements**

23.13 LXI Device Specific Configuration Schema

The LXI Device Specific Configuration represents device-specific or automatically configured network settings of the device network interface. If the device configuration enables automatic configuration, such as DHCP, any configuration specified in the LXI Device Specific Configuration Schema may be superseded.

RULE:23.13-1 Devices shall retain the LXI Device Specific configuration and only utilize it when automatic configuration is disabled. Thus, writing the LXI Device Specific Configuration while automatic configuration is active then disabling automatic configuration will result in the device using the configuration specified in LXI Device Specific Configuration.

Reading the LXI Device Specific Configuration from the device always returns the current settings of the interface over which it is read, regardless of if the settings were statically configured or received from automatic configuration.

To determine if automatic configuration is enabled read the LXI Common Configuration.

This schema specifies the XML namespace:

23.13.1 LXIDeviceSpecificConfiguration

LXIDeviceSpecificConfiguration contains various settings associated with the network interface that are potentially device specific.

For details on the various settings, see the LXI Device specification.

23.13.1.1 Attributes

Attribute	Syntax	LCI	Description
name	**Type:** xs:string **Card.:** Opt. **Default:** *None*	NA	*name* indicates the name of the interface described by this document. *name* is required on a GET and shall indicate the name used for the interface in the *LXICommonConfiguration Interface/@name* attribute. Devices with a single interface shall use the name "LXI". *name* is optional on a PUT. If absent, the interface over which this XML is delivered is configured. **OBSERVATION:** providing the *LXICommonConfiguration Interface/@name* here permits the client to associate the device specific IP configuration with the configuration in the *LXICommonConfiguration*. **Required: RULE:23.13.1.1-1** **Unsecure impact:** NA

23.13.1.2 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
IPv4Device	lxi:IPv4Device	Optional	*IPv4Device* contains the device-specific configuration related to IPv4. **RULE:23.13.1.2-1** LXI Devices shall accept *IPv4Device*. **RULE:23.13.1.2-2** If IPv4Device is absent, and the LXI Common Configuration does not specify automatic configuration, the IPv4 capability is disabled.
IPv6Device	lxi:IPv6Device	Optional	*IPv6Device* contains the device-specific configuration related to IPv6. **RULE:23.13.1.2-3** LXI Devices shall accept *IPv6Device* **RULE:23.13.1.2-4** If *IPv6Device* is absent, and the LXI Common Configuration does not specify any automatic configuration, the IPv6 capability is disabled.

Any element	*Any type*	Optional unbounded	Extension elements may be use to provide arbitrary interface configuration.

23.13.2 IPv4Device

IPv4Device represents the device-specific state of the IP version 4 capabilities of the device that are potentially device-specific.

When *IPv4Device* is written, the point in time at which it takes affect is device dependent.

23.13.2.1 Attributes

Attribute	Syntax	LCI	Description
address	**Type:** xs:string **Card.:** Opt. **Default:** *None*	No change	*address* contains the IPv4 address of the device. **Required: RULE:23.13.2.1-1** **Unsecure impact:** Any
subnetMask	**Type:** xs:string **Card.:** Opt. **Default:** *None*	No change	*subnetMask* contains the subnet mask to use. **Required: RULE:23.13.2.1-2** **Unsecure impact:** any
gateway	**Type:** xs:string **Card.:** Opt. **Default:** *None*	No change	*gateway* contains the gateway address. **Required: RULE:23.13.2.1-3** **Unsecure impact:** any
dns1	**Type:** xs:string **Card.:** Opt. **Default:** *None*	No change	*dns1* is the address of the first DNS server. **Required: RULE:23.13.2.1-4** **Unsecure impact:** any
dns2	**Type:** xs:string **Card.:** Opt. **Default:** *None*	No change	*dns2* is the address of the second (alternate) DNS server. **Required: RULE:23.13.2.1-5** **Unsecure impact:** any
Any Attribute	**Type:** *Any type* **Card.:** *Optional* **Default:** *NA*	NA	Arbitrary extension attributes may be included to provide device-specific IPv4 configuration that is beyond the scope of the LXI requirements. **RULE:23.13.2.1-6** LXI devices shall ignore extension attributes they do not recognize. **Required:** No **Unsecure impact:** NA

The IPv4Device complex type has **no subelements**

23.13.3 IPv6Device

IPv6Device represents the device-specific state of the IP version 6 capabilities of the device that are potentially device-specific.

When *IPv6Device* is written, the point in time at which it takes affect is device dependent.

23.13.3.1 Attributes

Attribute	Syntax	LCI	Description
Any Attribute	**Type:** *Any type* **Card.:** *Optional* **Default:** *NA*	NA	Arbitrary extension attributes may be included to provide device-specific IPv6 configuration that is beyond the scope of the LXI requirements. **RULE:23.13.3.1-1** LXI devices shall ignore extension attributes they do not recognize. **Required:** No **Unsecure impact:** NA

23.13.3.2 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
StaticAddress	lxi:IPv6Address	Optional unbounded	*StaticAddress* is optional and contains the device static address. **OBSERVATION:** *If the LXICommonConfiguration/Network/IPv6/@staticAddressEnabled is false, the static addresses are not used.* **RULE:23.13.3.2-1** Devices shall accept at least one *StaticAddress*.
Link Local Address	lxi:IPv6Address	Optional	*LinkLocalAddress* is a read-only field that contains the devices current link local address. **RULE:23.13.3.2-2** LXI Devices shall include the link local address in responses.
GlobalAddress	lxi:IPv6Address	Optional unbounded	*GlobalAddress* is a read-only element that contains the addresses provided to the device via router advertisement or DHCP. **RULE:23.13.3.2-3** A *GlobalAddress* element shall be included in the response for every device global address.

OBSERVATION: *Since unique-local may be determined by router advertisement or stateful DHCPv6 it is returned using a GlobalAddress element.*

23.13.4 IPv6Address

IPv6Address contains an IPv6 address.

23.13.4.1 Attributes

Attribute	Syntax	LCI	Description
address	**Type:** xs:string **Card.:** Req. **Default:** NA	NA	*address* contains the IPv6 address in CIDR notation. **Required: RULE:23.13.4.1-1** **Unsecure impact:** NA
router	**Type:** xs:string **Card.:** Opt. **Default:** *None*	NA	*router* contains the router IPv6 address if this *IPv6Address* has an associated router. The address is in CIDR notation. **Required: RULE:23.13.4.1-2** **Unsecure impact:** NA
dns	**Type:** xs:string **Card.:** Opt. **Default:** *None*	NA	*dns* contains the address of the IPv6 domain name server if this *IPv6Address* has an associated dns. The address is in CIDR notation. **Required: RULE:23.13.4.1-3** **Unsecure impact:** NA

The IPv6Address complex type has **no subelements**

23.14 LXI Certificate Reference Schema

The LXI Certificate schema indicates a single X.509 certificate, certificate chain, or CSR (Certificate Signing Request) that is on the device.

The certificate is not included in this schema, rather the entity on the device is identified using a GUID. The GUID is assigned by the device and is returned by the Certificate List API.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXICertificateRef/1.0, version: *1.0*
Editorial date:June 30, 2022

23.14.1 LXICertificateRef

23.14.1.1 Attributes

Attribute	Syntax		LCI	Description
GUID	**Type:**	xs:string	NA	The GUID identifies the certificate, certificate list, or CSR. The GUID is returned by the Certificate List API.
	Card.:	Req.		**Required: RULE:23.14.1.1-1**
	Default:	NA		**Unsecure impact:** NA

The LXICertificateRef complex type has **no subelements**

23.15 LXI Certificate List Schema

The LXI Certificate List schema represents a list of X.509 certificates, certificate chains, and CSR (Certificate Signing Requests) currently on the device.

The returned list of certificates includes a GUID that the client can use to delete the certificate.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXICertificateList/1.0, version: *1.0*
Editorial date:June 30, 2022

23.15.1 LXICertificateList

LXICertificateList contains a list of certificate entities on a device. Each is assigned a GUID that can be used to further manipulate the certificate.

The *LXICertificateList* complex type has **no attributes**

23.15.1.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
Certificate Info	lxi:CertificateInfo	Required unbounded	*CertificateInfo* contains information about a certificate on the device, including the GUID which may be used to operate on the certificate.

23.15.2 CertificateInfo

CertificateInfo contains information about a certificate, certificate list, or CSR (certificate signing request).

The GUID included in the *CertificateInfo* is used to manipulate the individual entity.

23.15.2.1 Attributes

Attribute	Syntax	LCI	Description

GUID	**Type:**	xs:string	NA	*GUID* is a Globally Unique Identifier generated by the device to represent this certificate. **Required: RULE:23.15.2.1-1** **Unsecure impact:** NA
	Card.:	Req.		
	Default:	NA		
Type	**Type:**	restriction of: xs:string	NA	*Type* indicates the kind of entity. One of the following values is returned: IDevID The entity is the Initial device identifier provided by the device manufacturer. LDevID The entity is a locally significant device identifier provisioned to the device by a user. CSR The entity is a Certificate Signing Request produced by the device to be signed by a certificate authority. **Required: RULE:23.15.2.1-2** **Unsecure impact:** NA
	Card.:	Req.		
	Default:	NA		
DNSName	**Type:**	xs:string	NA	*DNSName* is the DNS Name from the certificate. **Required: RULE:23.15.2.1-3** **Unsecure impact:** NA
	Card.:	Req.		
	Default:	NA		
Enabled	**Type:**	xs:boolean	NA	*Enabled* indicates if the corresponding certificate or certificate chain is enabled for use by the device. *Enabled* is meaningless for Certificate Signing Requests. *Enabled* shall be returned *true* for CSRs. **Required: RULE:23.15.2.1-4** **Unsecure impact:** NA
	Card.:	Req.		
	Default:	NA		
expiration Date Time	**Type:**	xs:string	NA	*expirationDateTime* is the expiration date and time of the certificate. For a CSR, *expirationDateTime* shall contain the requested expiration time from the CSR. If the CSR *LXICertificateRequest/ExpirationDateTime* was absent an empty string shall be returned. **RULE:23.15.2.1-5** The expiration date and time shall be expressed in ASN.1 format using ASN.1 GeneralizedTime per RFC5280. **OBSERVATION:** *The device will need to convert GeneralizedTime to UTC time if the year is between 1950 and 2050.* **Required: RULE:23.15.2.1-6** **Unsecure impact:** NA
	Card.:	Req.		
	Default:	NA		

The CertificateInfo complex type has **no subelements**

23.16 LXI Certificate Request Schema

The LXI Certificate Request schema is used by both the *getCSR* and *createCertificate* APIs for the client to specify attributes of the certificate it is requesting.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXICertificateRequest/1.0, version: *1.0*
Editorial date:June 30, 2022

23.16.1 LXICertificateRequest

LXICertificateRequest contains attributes that a client may request be used for a device certificate.

The *LXICertificateRequest* complex type has **no attributes**

23.16.1.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
SubjectName	lxi:SubjectName	Optional	*SubjectName* specifies the attributes of the distinguished name to be used in the subject of the certificate. The subject of the certificate indicates the identity of the LXI device.
AltDnsName	xs:string	Optional unbounded	*AltDnsName* specifies the alternate DNS name to be used in the certificate.
AltIPAddress	xs:string	Optional unbounded	*AltIPAddress* specifies the alternate IP Address to be used in the certificate. Multiple IP addresses may be specified as a comma separated list.
Expiration Date Time	xs:string	Optional	*ExpirationDateTime* indicates the time at which the requested certificate will expire. **RULE:23.16.1.1-1** The expiration date and time shall be expressed in ASN.1 format using ASN.1 GeneralizedTime per RFC5280. **OBSERVATION:** *The device will need to convert GeneralizedTime to UTC time if the year is between 1950 and 2050.*
Signature Algorithm	xs:string	Optional	*SignatureAlgorithm* specifies the signature algorithm that the certificate keyset should use. The string is the OID string specified in RFC 3279 or its hierarchy of successors. If absent the signature algorithm is device dependent.

			RULE:23.16.1.1-2 If the device does not support the requested signature algorithm, then the certificate request shall fail. The returned LXIProblemDetails/Title element shall contain an indication that the SignatureAlgorithm was invalid. The LXIProblemDetails/Instance shall have a comma separated list of accepted values. **OBSERVATION:** *Clients can determine the supported signature algorithms by sending the SignatureAlgorithm element with an empty string for the SignatureAlgorithm.*
Certificate Extension	lxi:CertificateExtension	Optional unbounded	*CertificateExtension* permits the user to request arbitrary certificate fields based on the object identifier and field values.

23.16.2 SubjectName

SubjectName contains the various attributes of the requested certificate subject.

RULE:23.16.2-1 The default fields for the subject name shall be the values used in the device IDevID.

The *SubjectName* complex type has **no attributes**

23.16.2.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
CommonName	xs:string	Optional	*CommonName* specifies the common name subject attribute.
Organization	xs:string	Optional	*Organization* specifies the organization subject attribute.
Organizational Unit	xs:string	Optional unbounded	*OrganizationUnit* specifies the organization unit subject attribute.
Locality	xs:string	Optional	*Locality* specifies the locality subject attribute.
State	xs:string	Optional	*State* specifies the state subject attribute.
Country	xs:string	Optional	*Country* specifies the country subject attribute.
SerialNumber	xs:string	Optional	*SerialNumber* specifies the serial number subject attribute.
Extra Subject Attribute	lxi:ExtraSubjectAttribute	Optional unbounded	*ExtraSubjectAttribute* specifies additional subject attributes not included in LXICertificateRequest using the Object ID and value.

23.16.3 ExtraSubjectAttribute

ExtraSubjectAttribute specifies an individual subject attribute.

The *ExtraSubjectAttribute* complex type has **no attributes**

23.16.3.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
ObjectID	xs:string	Required	*ObjectID* is the object ID that indicates the subject attribute as specified by the OpenGroup. The format of this string is a series of dot-separated integers. **RULE:23.16.3.1-1** *ObjectID shall be included.*
ObjectValue	xs:string	Required	*ObjectValue* is the subject value associated with the specified attribute. **RULE:23.16.3.1-2** *ObjectValue shall be included.*

23.16.4 CertificateExtension

The *CertificateExtension* complex type has **no attributes**

23.16.4.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
ObjectID	xs:string	Required	*ObjectID* is the object ID that indicates the certificate extension as specified by the OpenGroup. The format of this string is a series of dot-separated integers. **RULE:23.16.4.1-1** *ObjectID shall be included.*
Critical	xs:boolean	Optional	*Critical* indicates that this certificate extension is critical.
ObjectValue	xs:base64Binary	Required	*ObjectValue* is the subject value associated with the certificate field. **RULE:23.16.4.1-2** *ObjectValue shall be included.*

23.17 LXI Literals Schema

The *LXILiterals* schema contains a single element with optional arbitrary attributes. It is used to pass arbitrary data to a method. As such, it does not provide syntactic validation of parameters.

This schema is intended to be used by methods that require minimal parameters, and would derive very little benefit from schema-based syntactic validation.

Methods that utilize this schema must document the attribute names and types used.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXILiterals/1.0, version: *1.0*
Editorial date:June 30, 2022

23.17.1 LXILiterals

LXILiterals contains arbitrary attributes that can be used to pass parameters of arbitrary types and names to REST methods.

Methods that utilize this schema must document the attribute names and types used.

23.17.1.1 Attributes

Attribute	Syntax		LCI	Description
Any Attribute	**Type:**	*Any type*	NA	Each attribute has an arbitrary name with an arbitrarily typed parameter.
	Card.:	*Optional*		**Required:** Must be implemented as required for parameters used in the method for which this is a parameter.
	Default:	*NA*		**Unsecure impact:** NA

The LXILiterals complex type has **no subelements**

23.18 LXI Problem Details Schema

The LXI Problem Details schema provides detailed explanation from the device regarding HTTP operations that do not have an implicit response. Further detail could be an explanation of error conditions, or other device status regarding the invoked method.

If the HTTP response is OK (200), the *LXIProblemDetails* response is not required.

For some use cases, such as determining authentication requirements, it may be appropriate for a client to intentionally generate an HTTP error then use this structure and the response headers to determine the requirements to access the designated resource.

In such cases, the information in this element may be redundant with information also available from response headers.

RULE:23.18-1 Devices shall return the *LXIProblemDetails* when the LXI API generates 40X errors.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXIProblemDetails/1.0, version: *1.0*
Editorial date:June 30, 2022

23.18.1 LXIProblemDetails

The LXI ProblemDetails element contains the details related to an HTTP error.

The *LXIProblemDetails* complex type has **no attributes**

23.18.1.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
Title	xs:string	Required	High level description of the method result, consistent with the HTTP status code returned. **RULE:23.18.1.1-1** *Title* shall be included.
Detail	xs:string	Optional	Detail regarding the specific method status, for instance, the nature of a syntactic error.
Instance	xs:string	Optional	Detail specific to the issue. For instance, for a syntax error this could contain details used to isolate and correct the problem, such as the line number or specific reference to a flawed syntactic element.

23.19 LXI Pending Details Schema

The LXI Pending Details schema provides detailed explanation from the server regarding HTTP operations that return an HTTP status of 202. The HTTP status of 202 indicates that the operation is pending.

RULE:23.19-1 Schema-valid XML responses, as defined by this schema, shall be returned by devices to indicate pending operations.

OBSERVATION: Other sections of this specification require that devices return the *LXIPendingDetails* whenever an LXI API method returns a status of 202.

This schema specifies the XML namespace:

http://lxistandard.org/schemas/LXIPendingDetails/1.0, version: *1.0*
Editorial date:June 30, 2022

23.19.1 LXIPendingDetails

The LXI PendingDetails element contains the details related to why an operation is pending and permits the client to determine when it is completed.

The *LXIPendingDetails* complex type has **no attributes**

23.19.1.1 Sub-elements

The following must occur in this order:

Element	Type	Cardinality	Requirements
URL	xs:anyURI	Required	*URL* provides a URL at which the client can perform a GET to determine the status of the pending operation. **RULE:23.19.1.1-1** *URL* shall be included. **OBSERVATION:** *When querying the URL the client will either receive another operation pending response with another instance of this XML or a status of OK that indicates the operation is complete.*

User Action Required	xs:boolean	Required	*UserActionRequired* indicates if the operation is blocked waiting for a user action. For instance, a front panel operation or a device reboot.
Estimated Time To Complete	xs:integer	Optional	*EstimatedTimeToComplete* indicates the amount of time in seconds to complete the operation. *EstimatedTimeToComplete* shall be included if *@UserActionRequired* is false. *EstimatedTimeToComplete* shall be omitted if the device is awaiting a user action and the device does not know when it will occur.
Details	xs:string	Optional	*Details* provides an explanation of the operation that is pending, or why it is pending.