

In-Sight Platform API

API Version	Description
1.0	Released with In-Sight 24.3.0
1.1	Released with In-Sight 25.1.0: Add support for custom server certificates
1.2	Released with In-Sight 25.2.0: Add support for syslog forwarding TCP framing option
2.0	Released with In-Sight 25.3.0: Add new server certificates endpoints and change some existing API
3.0	Released with In-Sight 26.1.0: Add new backup, restore, and firmware upgrade endpoints

Document Revision 26.2.0.1

Table of Contents

- 1. Overview..... 3
 - 1.1 Lifecycle Definition..... 3
 - 1.1.1 Beta..... 3
 - 1.1.2 Released..... 3
 - 1.1.3 Stable..... 3
 - 1.1.4 Deprecated..... 3
 - 1.1.5 Removed..... 3
- 2. Platform API..... 4
 - 2.1 Root Resources (“api”)..... 4
 - 2.1.1 Audit Log..... 4
 - 2.1.2 Server Certificates..... 10
 - 2.1.3 Firmware Upgrade..... 13
 - 2.1.4 Backup and Restore..... 14
- 3. Object Types..... 14
 - 3.1 Audit Log..... 16
 - 3.1.1 Priority..... 16
 - 3.1.2 Structured Data..... 16
 - 3.1.2.1 Source..... 17
 - 3.1.2.2 Category..... 17
 - 3.1.3 Syslog Forwarding..... 17
 - 3.2 CertificateAndCsrMetadataListModel..... 18
 - 3.2.1 CertificateMetadataModel..... 18
 - 3.2.2 X509NameModel..... 18
 - 3.2.3 CSRDetailed..... 18
 - 3.2.4 ProtocolModel..... 19
- 4. Error Handling..... 19
 - 4.1 HTTP Responses..... 19

1. Overview

The In-Sight Platform API uses the HTTP protocol to access and change resource values on the camera. This document describes the Platform API that is available via that protocol.

1.1 Lifecycle Definition

Each Platform API provided by Cognex exists in one of the lifecycle states defined below. These states are intended to help you understand the level of stability, support, and future availability to expect from a given API.

1.1.1 Beta

A Beta API is an early-access feature provided for testing and feedback. It may change significantly or even be removed in future releases. If you choose to use a Beta API, be aware that your code may require significant changes when new API versions are released.

1.1.2 Released

A Released API is considered mature and ready for use in production environments. It is fully supported by Cognex and is intended for general use. However, future API versions may still introduce breaking changes, so it's important to review release notes and test your applications when updating them to use new API versions.

1.1.3 Stable

A Stable API is a Released API that comes with a guarantee of backwards compatibility. If any changes are needed, they will be introduced through a deprecation process, ensuring your existing usages continue to work. Stable APIs are the safest choice for long-term projects.

1.1.4 Deprecated

A Deprecated API is a formerly Released API that is now scheduled for removal in a future API version. Deprecated APIs remain available for at least 15 months after being marked as deprecated, giving you time to update your code before the API is removed.

1.1.5 Removed

A Removed API is no longer accessible in the product. Any code that depends on a Removed API will stop working after the API is removed, so it is important to migrate away from Deprecated APIs before they reach this state.

2. Platform API

In the following section, HTTP requests are made to resources.

2.1 Root Resources (“api”)

The following items’ base resource is “api” (e.g. “api/audit-log”).

Example usage with curl:

```
curl --user admin: --request GET http://127.0.0.1/api/audit-log
```

Default username is “admin” and password “”. The user must have an access level of “Full” to be authorized to make requests to these resources.

2.1.1 Audit Log

Endpoint	Description		
audit-log	Lifecycle State: Released		
	GET – Get the audit log.		
	Example usage with curl: curl --user admin: --request GET http://127.0.0.1/api/audit-log? before=Value&after=Value --header "Accept-Encoding: gzip"		
	Query Parameters		
	Parameter	Value	Description
	before	ISO 8601 formatted date	Only events with a timestamp before the provided date are included in the response. When omitted, there is no "before" filtering applied.
	after	ISO 8601 formatted date	Only events with a timestamp after the provided date are included in the response. When omitted, there is no "after" filtering applied.
	Supported Coding Types (Compression)		
	Request “Accept-Encoding” Header	Response “Content-Encoding” Header	Description
	"Accept-Encoding:" (or not given)	N/A	No encoding requested; response will yield uncompressed JSON data.

	"Accept-Encoding: gzip"	"Content-Encoding: gzip"	"gzip" encoding requested; response will yield compressed JSON data, which can be decompressed with "gzip -d" or similar.
Response Data JSON array literal of Audit Log objects. <pre> { "\$schema": "https://json-schema.org/draft/2019-09/schema", "title": "Download audit log, GET /api/audit-log", "type": "array", "items": { "description": "Download audit log event entry", "type": "object", "required": ["timestamp", "hostname", "severity", "type", "event", "result", "structuredData"], "properties": { "timestamp": { "description": "ISO 8601 formatted timestamp of the event", "type": "string", "format": "date-time" }, "hostname": { "description": "Hostname of the machine the event occurred on", "type": "string", "minLength": 1 }, "severity": { "description": "Log severity of the event", "type": "string", "enum": ["emerg", "alert", "crit", "err", "warning", "notice", "info", "debug"] }, "type": { "description": "Audit log event type", "type": "string", "minLength": 1 }, "event": { "description": "Audit log event id", "type": "string", "minLength": 1 }, "result": { "description": "Audit log event result", "type": "string", "minLength": 0 }, "structuredData": { "description": "Structured data in the corresponding syslog messages", "required": ["auditEvent@39975", "timeQuality"] } } } }</pre>			

```

    ],
    "properties": {
      "auditEvent@39975": {
        "description": "Additional IEC 62443 4-2 audit event data",
        "required": [
          "category",
          "source",
          "id"
        ],
      },
      "additionalProperties": false,
      "properties": {
        "source": {
          "description": "Actor which triggered the audit log event",
          "type": "string",
          "enum": [
            "device",
            "remoteSystem",
            "humanUser"
          ]
        },
        "category": {
          "description": "Audit log event category",
          "type": "string",
          "enum": [
            "accessControl",
            "requestError",
            "controlSystem",
            "backupRestore",
            "config",
            "auditLog"
          ]
        },
        "id": {
          "description": "Id of the audit event resetting to 0 each boot and incrementing
by 1 for each successive event",
          "type": "integer",
          "minimum": 0,
        }
      }
    },
    "timeQuality": {
      "description": "RFC 5424 timeQuality structured data parameters",
      "required": [
        "isSynced",
        "tzKnown"
      ],
      "properties": {
        "isSynced": {
          "description": "Identifies whether the timestamp was recorded while the device
was synchronized with a time server",
          "type": "string",
          "enum": [
            "0",
            "1"
          ]
        },
        "tzKnown": {
          "description": "Identifies whether the timestamp was set with the time zone
configured",
          "type": "string",
          "enum": [
            "0",
            "1"
          ]
        }
      }
    },
    "auth@39975": {
      "description": "Describe the authorization of an event - who did it",
      "required": [
        "user"
      ],
      "additionalProperties": false,
      "properties": {
        "user": {
          "description": "Id of coguser which was authorized for the event",
          "type": "string",

```

	<pre> "minLength": 1 } }, "configChange@39975": { "description": "Describe the configuration change", "required": ["old", "new"], "additionalProperties": false, "properties": { "property": { "description": "Description of the property being changed if the 'event id' is too constrained to adequately identify the system property being configured.", "type": ["string", "number", "integer", "object", "array", "boolean", "null"] }, "old": { "description": "Value used for the configuration option prior to the config change event", "type": ["string", "number", "integer", "object", "array", "boolean", "null"] }, "new": { "description": "New value used for the configuration option as a result of the config change event", "type": ["string", "number", "integer", "object", "array", "boolean", "null"] } } } } }</pre>
audit-log/ syslog- forwarding	Lifecycle State: Released GET – Get the current syslog forwarding configuration of whether the forwarding is enabled. When enabled, also include the address, port, and severity filter. Example usage with curl: curl --user admin: --request GET http://127.0.0.1/api/audit-log/syslog-forwarding Response Data <pre>{ "\$schema": "https://json-schema.org/draft/2019-09/schema", "title": "GET /api/audit-log/syslog-forwarding response data", "type": "object",</pre>

```

"required": [
  "enabled"
],
"additionalProperties": false,
"properties": {
  "enabled": {
    "type": "boolean"
  },
  "address": {
    "type": "string",
    "anyOf": [
      {
        "format": "idn-hostname"
      },
      {
        "format": "ipv4"
      }
    ]
  },
  "description": "Address of the remote syslog collector which audit events will be
forwarded to"
},
"port": {
  "type": "integer",
  "minimum": 1,
  "maximum": 65535
},
"severityFilter": {
  "description": "Log severity of the event",
  "type": "string",
  "enum": [
    "emerg",
    "alert",
    "crit",
    "err",
    "warning",
    "notice",
    "info",
    "debug"
  ]
},
"tcp_framing": {
  "description": "TCP framing mode",
  "type": "string",
  "enum": [
    "octet-counted",
    "traditional"
  ],
  "default": "octet-counted"
}
},
"allof": [
  {
    "if": {
      "properties": {
        "enabled": {
          "const": true
        }
      }
    },
    "then": {
      "required": [
        "severityFilter",
        "port",
        "address",
        "tcp_framing"
      ]
    }
  }
],
"examples": [
  {
    "enabled": false
  },
  {
    "enabled": true,
    "address": "example-server-hostname",
    "port": 6514,

```

```

        "severityFilter": "warning",
        "tcp_framing": "octet-counted"
    }
}
}

```

Lifecycle State: Released

PUT – Enable syslog forwarding by specifying the remote forwarding address and optionally an event severity filter. Note: Syslog forwarding requires octet-counting TCP framing in the syslog collector.

Example usage with curl:

```
curl --user admin: --request PUT --header "Content-Type: application/json" --data "Request Data"
http://127.0.0.1/api/audit-log/syslog-forwarding
```

Request Data

```

{
  "$schema": "https://json-schema.org/draft/2019-09/schema",
  "title": "PUT /api/audit-log/syslog-forwarding request data",
  "type": "object",
  "required": [
    "address"
  ],
  "additionalProperties": false,
  "properties": {
    "severityFilter": {
      "description": "Log severity of the event",
      "type": "string",
      "enum": [
        "emerg",
        "alert",
        "crit",
        "err",
        "warning",
        "notice",
        "info",
        "debug"
      ],
      "default": "info"
    },
    "port": {
      "type": "integer",
      "minimum": 1,
      "maximum": 65535,
      "default": 6514
    },
    "address": {
      "type": "string",
      "anyOf": [
        {
          "format": "idn-hostname"
        },
        {
          "format": "ipv4"
        }
      ],
      "description": "Address of the remote syslog collector which audit events will be forwarded to"
    },
    "tcp_framing": {
      "description": "TCP framing mode",
      "type": "string",
      "enum": [
        "octet-counted",
        "traditional"
      ],
      "default": "octet-counted"
    }
  },
  "examples": [
    {
      "address": "example-server-hostname",
      "port": 6514,

```

	<pre> "severityFilter": "info", "tcp_framing": "octet-counted" }] } </pre>
	Lifecycle State: Released DELETE – Disable syslog forwarding. Example usage with curl: curl --user admin: --request DELETE http://127.0.0.1/api/audit-log/syslog-forwarding

2.1.2 Server Certificates

Note: The Server Certificates API is currently under active development. **Breaking changes** are expected in the coming releases as we continue to enhance and refine its functionality.

The following endpoints allow users to create, read, and delete server certificates on the device. After installing a certificate, users must activate it by selecting one of the supported protocols available from the protocols endpoint.

Endpoint	Verb Description
security/certificates/protocols	Lifecycle State: Released GET – Get the list of supported protocols. Example usage with curl: curl --user admin: --request GET http://127.0.0.1/api/security/certificates/protocols Response Data 200 – JSON array of ProtocolModel objects
security/certificates/tls	Lifecycle State: Released GET – Get the list of installed certificates and pending certificate signing requests. Example usage with curl: curl --user admin: --request GET http://127.0.0.1/api/security/certificates/tls Response Data 200 – JSON object literal of CertificateAndCsrMetadataListModel
security/certificates/tls/rename	Lifecycle State: Released POST – Rename a certificate using two query params, src and dest. Example usage with curl: curl --user admin: --request POST "http://10.28.124.120/api/security/certificates/tls/rename? src=existing_cert_name&dest=new_cert_name "

	Response Data 200 – TLS certificate renamed 400 – Invalid request 404 – TLS certificate not found 409 – Certificate ID already in use 500 – Internal server error						
security/certificates/tls/{id}	Lifecycle State: Released GET – Get the certificate. Example usage with curl: <pre>curl --user admin: --request GET http://127.0.0.1/api/security/certificates/tls/INSIGHT_TLS_CERT</pre> Response Data 200 – JSON object literal of CertificateMetadataModel 404 – TLS certificate not found						
	PUT – Install a certificate with an encrypted private key. The private key must be password protected, and the password must be provided to unwrap the private key. <table> <tr> <td>cert_file required</td><td>string <binary> DER encoded certificate file</td></tr> <tr> <td>encrypted_private_key_file required</td><td>string <binary> Encrypted PKCS#8 private key file</td></tr> <tr> <td>password required</td><td>string Password for encrypted private key</td></tr> </table> Example usage with curl: <pre>curl --user admin: --request PUT --form cert_file=@/data/workspace/certificate-management/premade_cert.der --form encrypted_private_key_file=@/data/workspace/certificate-management/premade_key.der --form password=the_keys_password http://127.0.0.1/api/security/certificates/tls/INSIGHT_TLS_CERT</pre> Response Data	cert_file required	string <binary> DER encoded certificate file	encrypted_private_key_file required	string <binary> Encrypted PKCS#8 private key file	password required	string Password for encrypted private key
cert_file required	string <binary> DER encoded certificate file						
encrypted_private_key_file required	string <binary> Encrypted PKCS#8 private key file						
password required	string Password for encrypted private key						

	200 – Certificate installed 400 – Invalid request 409 – Certificate already installed with another ID, or ID already in use 500 – Internal server error
	Lifecycle State: Released DELETE – Delete the certificate. Example usage with curl: <pre>curl --user admin: --request DELETE http://127.0.0.1/api/security/certificates/tls/INSIGHT_TLS_CERT</pre> Response Data 204 – TLS certificate deleted 400 – Invalid TLS certificate identity 404 – TLS certificate not found
security/certificates/tls/{id}/csr	Lifecycle State: Released PUT – Generate a certificate signing request for the device using a 3072-bit RSA key. Example usage with curl: <pre>curl --user admin: --request PUT --header "Content-Type: application/json" --data '{"subject": {"C": "US", "ST": "MA", "L": "Natick", "O": "Cognex", "CN": "IS3805MP-abcdef"}, "subjectAltName": ["DNS:IS3805MP-abcdef", "DNS:IS3805MP-abcdef.net.corporation.com", "IP:127.0.0.1"]}' http://127.0.0.1/api/security/certificates/tls/INSIGHT_TLS_CERT/csr</pre> Response Data 200 - String containing the contents of a certificate signing request 400 – Invalid CSR identity
security/certificates/tls/{id}/pem	Lifecycle State: Released PUT – Install a signed certificate. Example usage with curl: <pre>curl -user admin: --request PUT --form cert_pem_file=@/data/workspace/certificate-management/signed.pem http://127.0.0.1/api/security/certificates/tls/INSIGHT_TLS_CERT/pem</pre>

	Response Data 200 – JSON object literal of CertificateMetadataModel 400 – Invalid request
security/certificates/tls/{id}/activate	Lifecycle State: Released PUT – Activate a TLS certificate for specific protocols or set it as the default certificate. Use an empty list to clear assignments, "default" to set the certificate as the default certificate, or a list of specific protocols to activate the certificate for. Note that this will cause a configuration change that may cause temporary downtime for some services. Example usage with curl: <pre>curl --user admin: --request PUT --header "Content-Type: application/json" --data '{ "protocols": ["https"] }' http://127.0.0.1/api/security/certificates/tls/INSIGHT_TLS_CERT/activate</pre> Response Data 200 – Certificate activated 400 – Invalid request 404 – TLS certificate not found 500 – Internal server error

2.1.3 Firmware Update

Note: The firmware update endpoint is accessible by users with Full access level.

Endpoint	Description										
firmware/update	Lifecycle State: Released POST – Apply a firmware update to the device. Example usage with curl: <pre>curl --user admin --request POST https://192.168.0.1/api/firmware/update?reboot=true --header "Content-Type: application/octet-stream" --data-binary "@/path/to/firmware.cogfw"</pre> Query Parameters <table><tr><th>Parameter</th><th>Type</th><th>Default</th><th>Description</th></tr><tr><td>reboot</td><td>Boolean</td><td>false</td><td>Indicates whether the device will reboot after a successful upgrade.</td></tr></table> Request Headers <table><tr><th>Name</th><th>Value</th></tr></table>	Parameter	Type	Default	Description	reboot	Boolean	false	Indicates whether the device will reboot after a successful upgrade.	Name	Value
Parameter	Type	Default	Description								
reboot	Boolean	false	Indicates whether the device will reboot after a successful upgrade.								
Name	Value										

Content-Type		application/octet-stream
Request Body		
Content-Type		Description
application/octet-stream		A valid .cogfw firmware file for the device.
Responses		
HTTPS Status Code	Description	
200	Success. The device has applied the supplied firmware.	
4xx	Upgrade failed from a client failure	
5xx	Upgrade failed from a server failure	

2.1.4 Backup and Restore

Note: Backup and restore endpoints are accessible by users with Full or Elevated access level.

Endpoint	Description												
backup-restore/backup	Lifecycle State: Released GET – Download a backup archive of the device configuration. Example usage with curl: curl --user admin --request GET https://192.168.0.1/api/backup-restore/backup --output "path/to/backup" Response Body <table><tr><th>HTTPS Status Code</th><th>Description</th><th>Content Type</th><th>Response Header</th></tr><tr><td>200</td><td>Success. Backup created successfully</td><td>application/octet-stream</td><td>X-Uncompressed-Size (integer): Uncompressed size of the backup contents, which can be used to estimate download progress</td></tr><tr><td>500</td><td>Backup creation failed</td><td>N/A</td><td>N/A</td></tr></table>	HTTPS Status Code	Description	Content Type	Response Header	200	Success. Backup created successfully	application/octet-stream	X-Uncompressed-Size (integer): Uncompressed size of the backup contents, which can be used to estimate download progress	500	Backup creation failed	N/A	N/A
HTTPS Status Code	Description	Content Type	Response Header										
200	Success. Backup created successfully	application/octet-stream	X-Uncompressed-Size (integer): Uncompressed size of the backup contents, which can be used to estimate download progress										
500	Backup creation failed	N/A	N/A										
backup-restore/restore	Lifecycle State: Released POST – Restores system state from a previous backup. A reboot may occur during this process. Example usage with curl: curl --user admin --request POST https://192.168.0.1/api/backup-restore/restore?replicate --header "Content-Type: multipart/form-data" --form "archive=@/path/to/backup" Query Parameters <table><tr><th>Parameter</th><th>Type</th><th>Description</th></tr><tr><td>type</td><td>string</td><td>Indicates the type of backup to perform. Options are "replicate" or "replace"</td></tr><tr><td colspan="3">Replication - restores all settings from the backup to</td></tr></table>	Parameter	Type	Description	type	string	Indicates the type of backup to perform. Options are "replicate" or "replace"	Replication - restores all settings from the backup to					
Parameter	Type	Description											
type	string	Indicates the type of backup to perform. Options are "replicate" or "replace"											
Replication - restores all settings from the backup to													

			the device, except for device-unique data such as network/hostname settings
			Replace - restores all settings from the backup to the device, including device-unique data such as network/hostname settings
	Request Headers		
	Name		Value
	Content-Type		multipart/form-data
	Request Body		
	Content-Type	Value	Description
	multipart/form-data	archive – binary	The binary archive of the backup to be restored to the device.
	Responses		
	HTTPS Status Code	Description	
	200	Success. The backup file has been restored to the device	
	400	Restoration failed	
	409	Restoration failed due to an ongoing restore operation	
	500	Restoration failed due to internal error	
	507	Restoration failed due to insufficient space on device	

3. Object Types

3.1 Audit Log

Property	Description
severity	Priority
type	Customer facing ID for the subsystem or process on the system that records the event.
event	The event ID uniquely identifies an audit log event within the scope of the audit log event type.
result	When an event ID may not sufficiently describe the result of an event due to non-static data, additional data may be provided in the event result. Events are allowed to have an empty event result.
structuredData	Structured Data
hostname	Name of the device on which the event occurred.
timestamp	System time in milliseconds when the log event was reported.

3.1.1 Priority

Level	Value	Example Usage
Emergency	emerg	N/A
Alert	alert	- Detected tamper events - Detected configuration incompatible with security policy
Critical	crit	N/A
Error	err	Failure to apply configuration changes
Warning	warning	Access denied to device
Notice	notice	Changes in system configuration
Informational	info	Access granted to device
Debug	debug	N/A

3.1.2 Structured Data

SD-ID	PARAM-NAME	PARAM-VALUE
timeQuality	tzKnown	See RFC-5424: The Syslog Protocol, section 7.1.1 tzKnown.
	IsSynced	See RFC-5424: The Syslog Protocol, section 7.1.2 isSynced.
auditEvent@39975	source	Source
	category	Category
	id	Monotonically increasing counter to track all successfully recorded audit events.
auth@39975	user	ID of coguser that was authorized for the event.

configChange@39975	property	Description of the property being changed if the event ID is too constrained to adequately identify the system property being configured.
	old	Value used for the configuration option prior to the config change event.
	new	New value used for the configuration option because of the config change event.

3.1.2.1 Source

Name	Value	Actor triggering the event
Originating Device	device	A process internal to the system
Software Process	remoteSystem	A remote system without human interaction
Human User Account	humanUser	A human interaction with the system

3.1.2.2 Category

Name	Value	Priority	Event Trigger
Access Control	accessControl	1	An access request to the device by any external person or system has been granted or denied. - Password/certificate-based authentication success/failure to the device - Access to the device granted/denied due to user permissions
Request Error	requestError	6	An error occurred while processing a request from a remote actor. Invalid request data
Control System Event	controlSystem	4	The device sends a request to a remote system to alter the behavior of that remote system.
Backup and Restore Event	backupRestore	5	- A backup download started - A backup download failed - A restore started - A restore completed - A restore failed
Configuration Change	config	3	The configuration of the system has been changed. - Network interface IP address changes - Common service configuration changes
Audit Log	auditLog	2	Any event that impacts the quantity, quality, or availability of information available in the audit log.

3.1.3 Syslog Forwarding

Property	Type	Description
enabled	boolean	Whether or not syslog forwarding is enabled. If enabled is false, then no other properties are returned in the GET request.
address	string	The address of the server to forward logs to.
port	integer	The port of the server to forward logs to.
severityFilter	string	The minimum severity threshold for log forwarding, where messages matching or exceeding this severity level are forwarded while less severe messages are excluded.
tcp_framing	string	The TCP framing mode to use, accepting either "octet-counted"

		(default) or “traditional” framing methods to determine message boundaries.
--	--	---

3.2 CertificateAndCsrMetadataListModel

Property	Type	Description
certificates	CertificateMetadataModel []	Array of certificate information.
csrs	CSRDetails []	Array of certificate signing request details. These are pending requests that will become certificates upon upload of a PEM file to this record.

3.2.1 CertificateMetadataModel

Property	Type	Description
id	string	Name given to the certificate for internal reference.
serialNumber	integer	Big-endian hexadecimal string set by the Certificate Authority.
version	integer	X509 version of the certificate.
subject	X509NameModel	Distinguished name of the client to which the certificate belongs.
issuer	X509NameModel	Distinguished name of the issuing Certificate Authority that signed the certificate.
notBefore	date-time (string)	ASN1_TIME of when the certificate becomes valid.
notAfter	date-time (string)	ASN1_TIME of when the certificate expires.
subjectAltName	array<string>	List of additional host names (IP addresses, common names, etc.) to be protected by this certificate. Values must be prefixed with IP: or DNS: depending on the type.
subjectHash	string	Hash value of the subject of the certificate.
expired	boolean	Whether or not the certificate has expired.

3.2.2 X509NameModel

Property	Type	Description
C	string	Two-letter ISO country code (e.g., US, CA)
ST	string	Geographic region within country
L	string	City or town name
O	string	Legal entity name (e.g., Example Corporation)
OU	string	Department or division (e.g., IT Department)
CN	string	Device name or device IP

3.2.3 CSRDetails

Property	Type	Description
id	string	Name given to the certificate signing request for internal reference.
subject	X509NameModel	X509 version of the certificate.

subjectAltName	string[]	List of additional host names (IP addresses, common names, etc.) to be protected by this certificate. Values must be prefixed with IP: or DNS: depending on the type.
----------------	----------	---

3.2.4 ProtocolModel

Property	Type	Description
name	string	The name of the protocol.
usesDefault	boolean	Whether or not this protocol is using the default certificate. False means that it's assigned to a user-uploaded certificate.

4. Error Handling

When there is an error completing the request, an error response is returned:

```
{
  "status": <http error response code>,
  "detail": "Example description of a user displayed error message for the bad request"
}
```

4.1 HTTP Responses

If you are making HTTP requests for the resources, then the expected response would look like this...

A successful GET:

```
curl --user admin: http://127.0.0.1/api/audit-log
200 OK Response with JSON payload in the body
```

When a page not found:

```
curl --user admin: http://127.0.0.1/api/audit-log2
HTTP ERROR 404 Not Found
<html>
<head><title>404 Not Found</title></head>
<body>
<center><h1>404 Not Found</h1></center>
<hr><center>nginx</center>
</body>
</html>
```

Access to a resource with unauthorized user:

```
curl --user John: http://127.0.0.1/api/audit-log
HTTP ERROR 401 Unauthorized
<html>
<head><title>401 Authorization Required</title></head>
<body>
<center><h1>401 Authorization Required</h1></center>
<hr><center>nginx</center>
</body>
</html>
```