D-TECT IP

Network integration & HTTP API

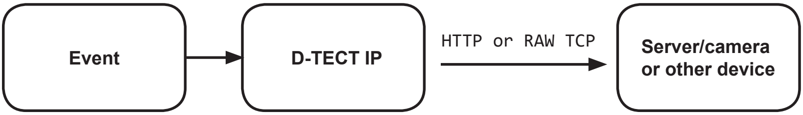
1. Introduction

The D-TECT IP can be set to send network alarms at different events, but to fully integrate and control its functions it is necessary to use the HTTP API.

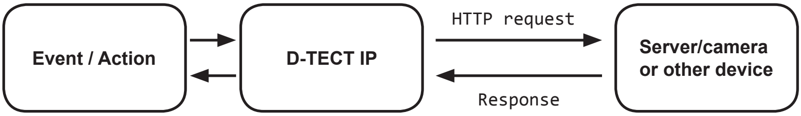
When using user configurable network alarms the VMS systems and IP cameras are set to listen for specific messages (as for example a HTTP API for an IP camera or a generic event input in a VMS). Event conditions can be set in the unit to only send alarms that are of interest. For almost any kind of event this includes alarm delay and timeout, but it can also be different types of threshold levels.

The HTTP API are used on all D-TECT IP units and allows control of all functions . This method allows all devices that can perform a HTTP request to integrate the D-TECT IP.

Integration using user configurable network alarms



Integration using HTTP API



2. User configurable network alarms

The D-TECT IP can perform user configurable network alarms at a number of different event triggers. It can be configured to a maximum of 40 individual network alarms according to the following table.

Number	Event	Condition	Condition Value	Trigger
1 ... 20	PIR Detection	Digital input		start
				stop
	Tampering	Accelerometer		start
				stop
	Low light detection	Lux level	2 ... 50	start
				stop
	Timer/heartbeat	Timer period (s)	1 ... 100000	start
				stop
	Temperature	Upper limit (ºC)	-40 ... 100	start
		Lower limit (º)		stop

Supported protocols: HTTP request (GET or POST) or RAW TCP
Supported data types: JSON, XML, plain text
Maximum URL length: 400 characters

3. HTTP API to control D-TECT

The following commands are used to control the D-TECT and to read sensor status. The HTTP API can be used by software or other IP devices.

All HTTP requests are of the GET method. The HTTP requests are based on the following layout:

```
http://<user>:<pass>@<ip-address>/<product>/<group>/<index>/<function>/<action>
```

`<user>` Saved username in the product
`<pass>` Saved password in the product
`<ip-address>` IP address of the product

3.1. D-Tect API function list

Product	Group	Index	Function	Action	Description
detect					Returns all status of the D-TECT
	sensor	pir_detection			Returns status of PIR
			value		Returns detection value of PIR
		low_light_ detection			Returns status of low light detector
			value		Returns value of light detector
		walk_test			Returns status of walk test
			state		Returns state walk test
				on	Sets walk test to On
				off	Sets walk test to Off
		temperature			Returns status of temperature sensor
			value		Returns temperature value
			unit		Returns temperature unit
	digital_ outputs				Returns status of all digital outputs
		1 ... 2			Returns status of selected digital output
			State		Returns state of selected digital output
				open	Sets selected digital output to open.
				close	Sets selected digital output to closed.
	tampering				Returns status of all tampering sensors
		lid			Returns status of selected tampering sensor
			state		Returns state of selected tampering sensor
				reset	Resets the state of selected tampering sensor. (automatic reset after 10 seconds)

3.2. Examples - Get status information

Each time a status is requested the D-TECT returns a JSON object. The JSON object is built up with the logic as described in section 3.1.

Get state of a specific digital output with the following request.

HTTP request: `http://user:pass@<product-ip>/dtect/digital_outputs/1/state`

Response: `“open” (alternatively “closed”)`

Get temperature status with the following request.

HTTP request: `http://user:pass@<product-ip>/dtect/sensor/temperature/value`

Response: `25.3`

Get complete status with the following request.

HTTP request: `http://user:pass@<product-ip>/dtect`

Response:

```
{
        “sensor”: {
                “pir_detection”: {
                                “value”: “no_alarm”
                },
                “low_light_detection”: {
                        “value”: 2
                },
                “walk_test”: {
                        “value”: “off”
                },
                “temperature”: {
                        “value”: 19.2,
                        “unit”: “&deg;C”
                }
        },
        “digital_outputs”: {
                “1”: {
                        “state”: “closed”
                },
                “2”: {
                        “state”: “open”
                }
        },
        “tampering”: {
                “lid”: {
                        “state”: “alarm”
                }
        }
}
```

3.3. Examples - Action commands

Close a specific digital output with the following request.

HTTP request: `http://user:pass@<product-ip>/dtect/digital_outputs/1/state/close`

Response: `OK`

Reset a specific tampering alarm with the following request.

HTTP request: `http://user:pass@<product-ip>/dtect/tampering/lid/state/reset`

Response: `OK`

4. Outgoing URL requests

The D-Tect IP can connect to other devices on the network by adding a “Connect to URL” action. POST and PUT requests may have a request body, written inside { } brackets.

4.1 Syntax

HTTP GET	http://user:pass@ip:port/path?argument1=value1&argument2=value2
HTTP POST	http_post://user:pass@ip:port/path?arg1=val1{arg2=val2}
HTTP PUT	http_put://user:pass@ip:port/path?arg1=val1{arg2=val2}
HTTP DELETE	http_delete://user:pass@ip:port/path?argument1=value1&argument2=value2
TCP/TELNET	tcp://ip:port{payload}

4.2 Argument substitution

%u	Uptime
%t	Temperature
%l	Ambient light in lux
%di1	PIR sensor status, “high” or “low”
%do1	Output 1 status, “open” or “closed”
%do2	Output 2 status, “open” or “closed”

4.3 Content-type

When making a POST or PUT request it is possible to change the content-type. The default content-type is text/plain. Content-type may be defined inside [ ] brackets.

HTTP POST	http_post://user:pass@ip:port/path{<tag>data</tag>}[text/xml]
HTTP PUT	http_put://user:pass@ip:port/path{{“key”:”val”}}[application/json]