Integrate with Nx Witness by Nx Server API

Nx TPE Technical Support Team

PREFACE

The Nx Witness™ Server API provides a standard set of API calls for 3rd party systems. With Nx Witness Server API, the vendors can duplicate a same log record, and send to Nx Witness via HTTP as generic event source.
On Nx Witness, we can setup the receiving rules and corresponding actions. When POS Log arrives at Nx as the generic event, you can display these logs or insert the log to timeline as a bookmark.

Used Protocol

Standard HTTP Protocol, RESTFUL API and GET Method to pass the parameters.
The name of Nx Witness API :/api/createEvent

Parameters

Parameters	Description	Required	Value
timestamp	Event date and time. If eventTimestamp is absent, current server date and time is used.	FALSE	UTC millisecond or ISO 8601 forma
source	External device name. It could be used as a filter at business rules to match different action with different devices. Also, user could see device name in the notifications panel. This parameters will be displayed in the event list in Nx Witness on Source column.	False (*)	Alphabet or numbers. Special chars are not recommend to use, but space(blank) is OK. The text are all encode by UTF-8.

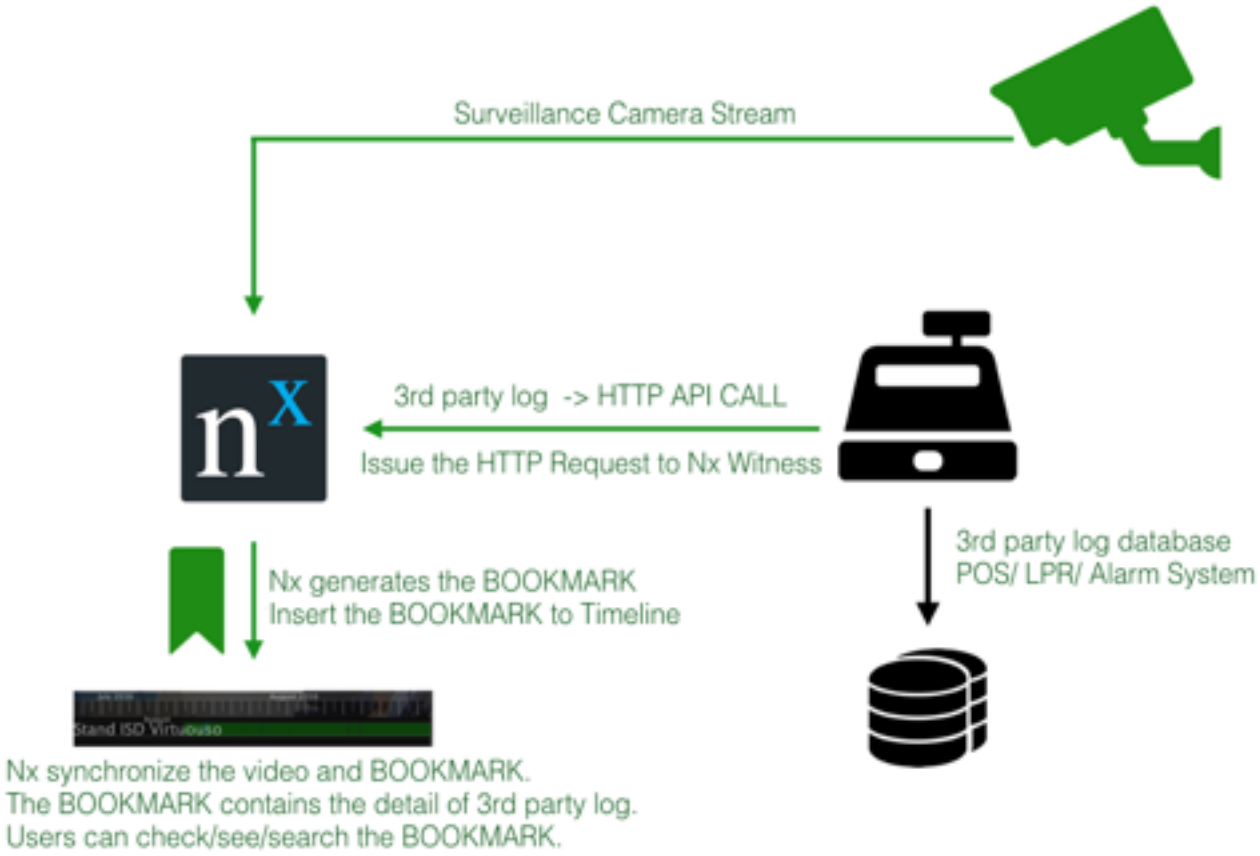
caption	Short event description. It could be used as a filter at business rules to math different action with different events. For example, if you are integrating with alarm system, you can put the event name on this filed, like FIRE ALARM. Or you are using the POS system, this field could be the name of the operator.	False (*)	Alphabet or numbers. Special chars are not recommend to use, but space(blank) is OK. The text are all encode by UTF-8.
description	Long event description. It could be used as a filter at business rules to math different action with different events. For example, if you are integrating with POS system, you can put the transaction detail in the filed, like order list, total amount, exchange…etc, or any critical event, such as cashbox opened.	False (*)	Alphabet or numbers. Special chars are not recommend to use, but space(blank) is OK. The text are all encode by UTF-8.
state	Adavnced usage. Generic events can be used either with long actions like 'do recording' or instant actions like 'send email'. If you are going to use long actions with Generic events, this parameter should be filled. For example, you are integrating the alarm system, you can setup the event which is send from alarm system to keep the active state for three minutes. Then you can send the first command as state is active, after 3 minutes, you send to second commend to set the state to inactive.	FALSE	Active / Inactive
(*) Though source,caption, and description are not required, however, in this API, one of them should always existed. Meaning that you are asked to assign at least one of them in the API call.			

HTTP Return Value

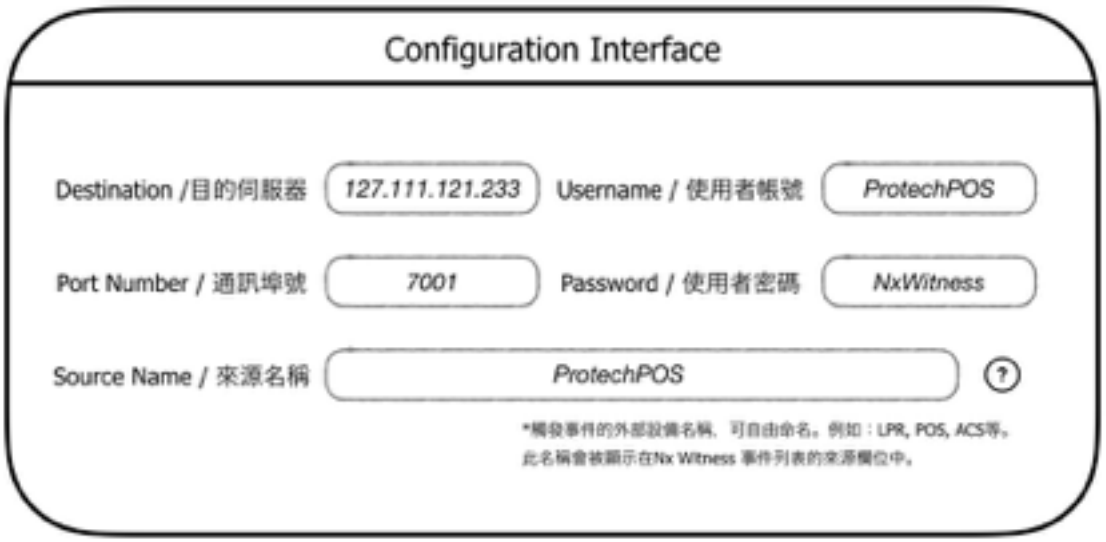
Return Json Object, including Error Code and Error Detail。

Success : Error Code = 0, errorString = null.
Fail : Error Code != 0, errorString = the reason why the request failed.

Data Flow


Surveillance Camera Stream
3rd party log -> HTTP API CALL
Issue the HTTP Request to Nx Witness
3rd party log database
POS/ LPR/ Alarm System
Nx generates the BOOKMARK
Insert the BOOKMARK to Timeline
Nx synchronize the video and BOOKMARK.
The BOOKMARK contains the detail of 3rd party log.
Users can check/see/search the BOOKMARK.

Configuration Interface on POS (Example)



Recommend Solution

1. The API should be using when you want to send any log to Nx Witness. (HTTP GET Method)
2. When the integrated system is writing internal Log or event history, meanwhile, the system could transit their log or event as a HTTP request, then send to Nx Witness.
3. At integrated system, it is required that to provide an user interface to configure the HTTP request targeted destination. The parameter should contains Nx Server IP Address, Port Number, Username, and Password。
4. Call the Nx API and append with correct parameters.
5. API Format : http://{Nx Witness Server IP}:{Port}/api/createEvent?{parmater-$1}={value} &{parmater-$2}={value}&……

Implementation Example

POS Log Sample

```
2016-06-27 13:17:20 Andy Index:A001
2016-06-27 13:17:23 Andy Tea Amount:1
2016-06-27 13:17:25 Andy Coke Amount:5
2016-06-27 13:17:26 Andy Milk Amount:5
2016-06-27 13:17:28 Andy Water Amount:2
2016-06-27 13:17:34 Andy Water Delete:1
```

When the log is generated, the integrated system can send out the HTTP request out at the same time, so the time and detail will be sychornized on both POS and NX Witness.

Nx Witness API

```
http://127.0.0.1:7001/api/createEvent?source=ProtechPOS&caption=Andy&description=Index:A001
http://127.0.0.1:7001/api/createEvent?source=ProtechPOS&caption=Andy&description=Tea Amount:1
http://127.0.0.1:7001/api/createEvent?source=ProtechPOS&caption=Andy&description=Coke Amount:5
http://127.0.0.1:7001/api/createEvent?source=ProtechPOS&caption=Andy&description=Milk Amount:5
http://127.0.0.1:7001/api/createEvent?source=ProtechPOS&caption=Andy&description=Water Amount:2
http://127.0.0.1:7001/api/createEvent?source=ProtechPOS&caption=Andy&description=Water Delete:1
```

To using the API call of Nx WItness, it always need to pass the authentication. Nx Witness supports two type of authentication, HTTP Basic / HTTP Digest .
We are recommending to use HTTP Digest, which much popular and safer.
Most of the programming language now have plenty of Framework to handle the authentication, so no need to implement by yourself.

SAMPLE CODE (python)

```
#Main Entry.

if __name__ == "__main__":

    serverIp = 'http://demo.networkoptix.com:7001'
    username = 'admin'
    password = 'yourpassword'
    get_params = {'source': 'ProtechPOS', 'caption': 'EmployeeName','description': 'POS LOG'}

    #print "sendGetReq ->"
    resp_get = sendGetReq(serverIp, '/api/createEvent', username, password, get_params)
```