

[可以教我如何實作 ONVIF 通訊協定嗎](#)

[可以提供詳細的 ONVIF XML 對話，以當做範例讓我進行實作嗎？](#)

[什麼是 Event Template？為何需要此參數？](#)

[此份文件有參考到哪些 ONVIF 規範？](#)

[我應該如何看待 ONVIF 實作線索](#)

[使用條件](#)

[XML namespaces \(XML 命名空間\)](#)

[圖示及顏色說明](#)

[如過我將要使用 Genius Vision NVR 來驗證我的攝影機，我需要具備什麼基本知識以操作此軟體？](#)

[什麼是 NVR 軟體中的「Channel」參數？作用為何？](#)

[攝影機的 ONVIF 初始連結取得影像串流程序為何](#)

[指令呼叫序列](#)

[必要指令簡介](#)

[非必要指令簡介](#)

[如何實作雙重串流 dual-streaming](#)

[為什麼需要雙重串流 dual-streaming?](#)

[實作](#)

[注意事項](#)

[如何實作設定編碼器參數 VideoEncoderOptions \(解析度,分辨率,Codec,幀率, FrameRate 等等 etc\)](#)

[如何實作設定影像參數 ImagingOptions \(亮度, 飽和度, 等\)](#)

[為什麼需要 ImagingSettings 的資訊](#)

[如何實作 PTZ 或快速球控制](#)

[指令呼叫序列](#)

[如何在 ONVIF 的 spec 下實作及支援事件通知 event notification](#)

[Genius Vision NVR 有支援 basic notification 嗎](#)

[如何實作 ONVIF Pull-point style 事件通知 event notification \(ET01\)](#)

[指令呼叫序列](#)

[如何實作 ONVIF motion detection \(位移偵測\) 事件 \(ET01\)](#)

[指令呼叫序列](#)

[事件格式定義](#)

[通知事件的 XML 範例](#)

[在 ONVIF 的 event 術語中，什麼是「Property」？什麼是「SetSynchronizationPoint」？要如何實作？](#)

[關於「Property」在規範中的介紹](#)

[Property 操作](#)

[什麼是 Synchronization Point \(同步點\)](#)

[NVC \(用戶端\) 需要主動呼叫 SetSynchronizationPoint 嗎](#)

[究竟什麼是 Synchronization Point\(同步點\)？我還是不明白](#)

[如果 NVC 不需要呼叫 SetSynchronizationPoint 指令，那麼這指令存在的意義為何？](#)

[何謂 notification streaming interface 事件通知？](#)

[如何實作 ONVIF 的警報輸入 Digital Input \(ET01\)](#)

[指令呼叫序列](#)

[取得警報輸入 digital input 的狀態變化](#)

[如何實作 ONVIF 繼電器輸出控制 Relay Output \(ET01\)](#)

[指令呼叫序列](#)

[取得繼電器輸出控制 relay output 的狀態變化](#)

[控制繼電器輸出 relay output 改變狀態](#)

[什麼是 InitialTerminationTime 爭議？如何實作](#)

[規範中的線索](#)

[可能的爭議](#)

[NVR 支援狀態](#)

[技術細節](#)

## 可以教我如何實作 ONVIF 通訊協定嗎

直接一對一教導如何實作 ONVIF 的細節實在不屬於我們「免費 ONVIF 獨立驗證服務」的範疇。

不過如果您願意，您可以聘請我們當做您的專家顧問，我們可以提供各式服務，但聘請我們進行顧問服務並不是免費的，請參照我們官方的[顧問服務網頁](#)。

在本文件的其餘部分，我們已經提供相當多線索，供您參考以面對實作方面會遇到的各種問題，這些資料應該已經足夠讓您可以大約摸索出 ONVIF 實作的全貌，但請注意：對於這些資料，我們[並不提供官方支援](#)。

## 可以提供詳細的 ONVIF XML 對話，以當做範例讓我進行實作嗎？

請參閱

- [可以教我如何實作 ONVIF 通訊協定嗎](#)

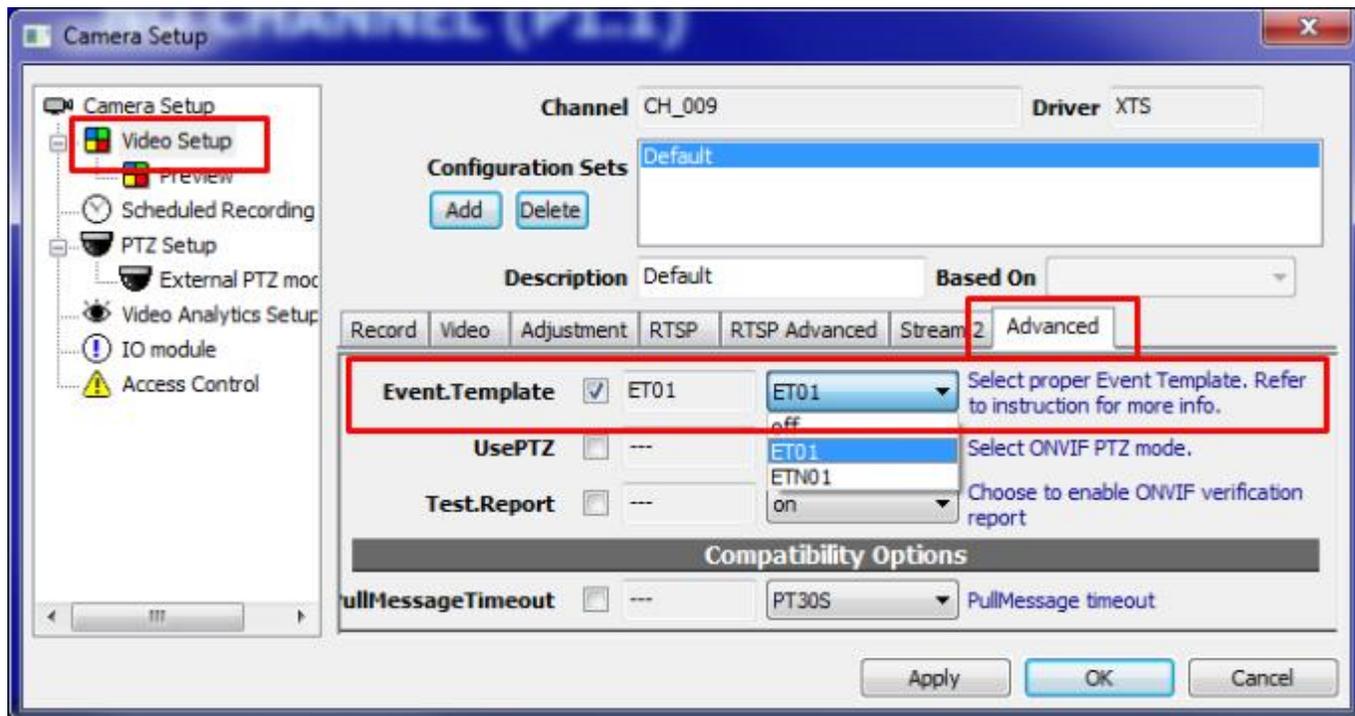
## 什麼是 Event Template？為何需要此參數？

**Event Template** 是 Genius Vision 所定義的一個實作參數，用以解決 ONVIF 規範定義不清楚時，開發者所可能面臨的問題：因 ONVIF 規範不明確，不同人會有不同解釋，實務上會造成無法相容。

根據[此處所述之原則](#)，也因為世上沒有一種規範可以寫得 100%明確而無模糊矛盾之地帶，我們加上此參數，以解決並分類一些常見的「不同解釋」，因而達到設備間相容的目的。

每一個 **Event Template** 的簡要定義整理可以在[此處參閱](#)。

以下畫面展示如何設定監看頻道的 **Event Template**。



此份文件有參考到哪些 ONVIF 規範？

- [ONVIF-Imaging-Service-Spec-v221.pdf](#)
- [ONVIF-Core-Specification-v230.pdf](#)
- [ONVIF-DeviceIo-Service-Spec-v221.pdf](#)

我應該如何看待 ONVIF 實作線索

使用條件

為了省去 IP 攝影機開發人員必須到處搜尋超過一打的 ONVIF 規範文件（甚至有些文件並不在 ONVIF 的範圍內），我們將過去專業的開發經驗整理成各式 ONVIF 實作線索，讓開發人員可以很容易在閱讀後知道如何實作。但您必須了解以下原則，作為使用本文件的先決條件：

1. 這些 ONVIF 實作線索並不享有**官方支援**，您若需要支援，[可聘請本公司當顧問](#)，顧問服務僅提供放給本公司標準產品的大量經銷商，而且需要另外支付顧問費用。
2. 這份文件內所述的內容，我們不保證具備任何的精確性，閱讀者需自行負擔因資訊錯誤而導致之風險。
3. ONVIF 的驗證「**不應該**」是透過解讀 XML 或 test report 達成，**而應該是用手動的方式測試，並用目測稽核測試結果是否符合預期**。我們已知許多 ONVIF 的功能缺陷，是導因於開發者過度依賴所謂的「test tool」或「test report」，我們必須認知「**驗證是人的工作，而非電腦（或程式）的工作，而我們不應該交付電腦人所應負擔的責任**」，特別是在一種市場的氣氛下，工程師知道如何透過「欺騙」test tool 而達到「符規」的目的。請[參閱此篇文章](#)取得更多資訊。
4. 所有的實作線索，都無法取代 ONVIF 的原始規範，我們仍然深信最好的學習 ONVIF 實作的方式，就是「**熟讀所有的 ONVIF 規範**」。
5. 如果您需要一個功能完整的測試軟體，我們很歡迎您下載本公司的試用版[全功能錄影圖控軟體](#)。
6. [這裡說明了](#)哪些 ONVIF 規範文件有被本文件參考到。

## XML namespaces (XML 命名空間)

- 除非文件內有另外定義，所有的 XML namespace prefix 都與規範「[ONVIF-Core-Specification-v230.pdf, 5.3 Namespaces](#)」中所述的一致。

### 圖示及顏色說明

- 在一些 XML 的範例中，如果您看到**綠色亮底字**，這代表您必須將該字串置換成適切的程式值。
- 我們會使用如「<ver20/imaging>**GetOptions**」的表示法，這意思是指「Imaging service 2.0 中的 **GetOptions** 指令」，在 ONVIF 規範中，這個指令與 Imaging service 1.0 中的 **GetOptions** 是**不同的指令**。這種表示法可以很清晰的表達在 ONVIF 規範中因為 XML namespace 的不同而導致的微小差異。更多資訊請參閱：[一般所謂 ONVIF 1.0, 1.1, 1.2, 2.0 是什麼意思？為什麼 1.0, 1.1, 1.2 及 2.0 會無法相容？](#)
- 黑色粗體字，例如「**GetProfiles**」通常指的是 ONVIF 規範中的指令名稱，而綠色粗體字，例如「**Channel**」則通常是表示 NVR 軟體中的設定值或內部變數，用以控制實際的 ONVIF NVC 軟體行為。

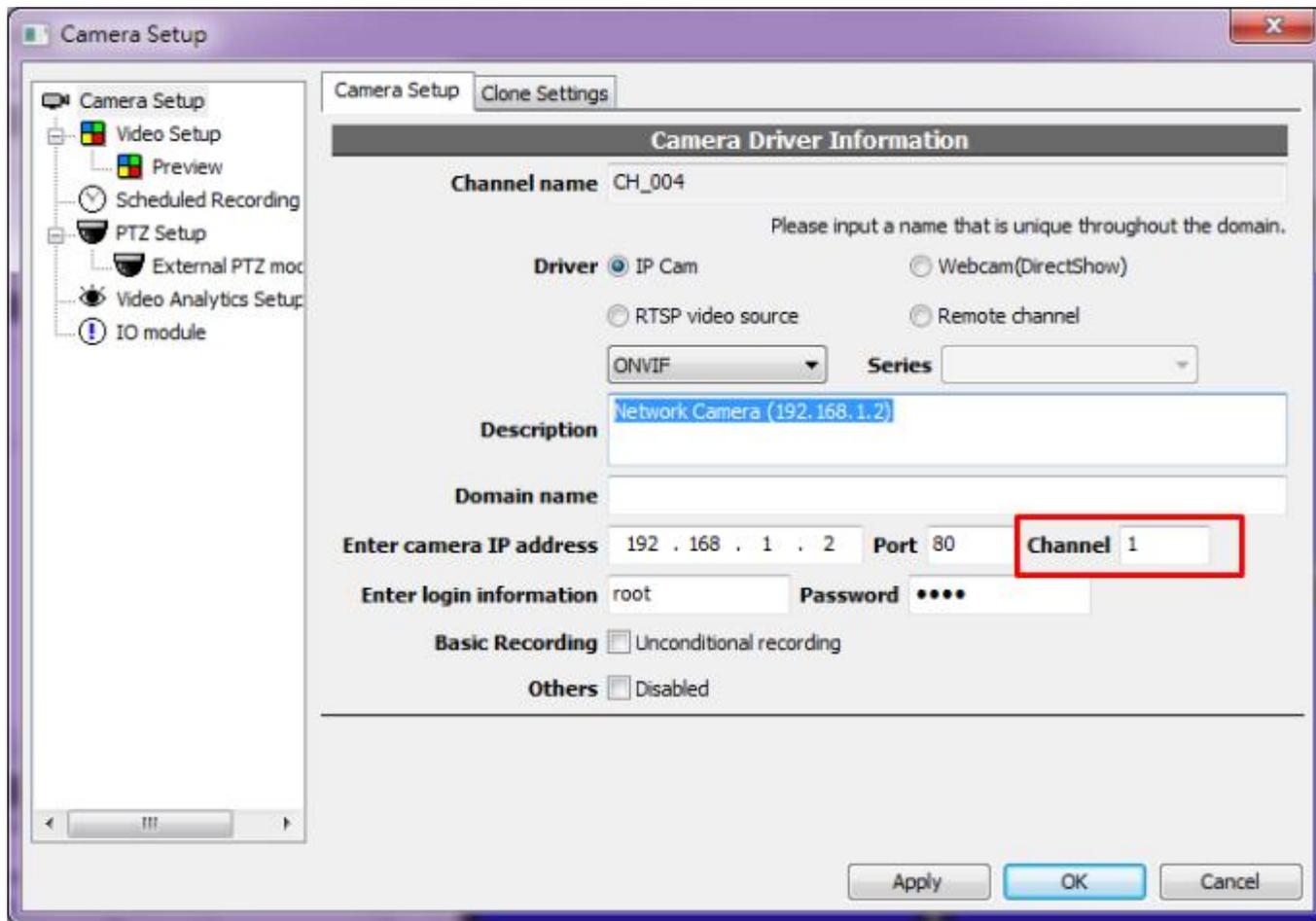
如過我將要使用 **Genius Vision NVR** 來驗證我的攝影機，我需要具備什麼基本知識以操作此軟體？

我們為了這個目的寫了一個精簡版本的使用手冊，請參閱以下連結：

- [Setup Genius Vision NVR for ONVIF Testing](#)

什麼是 NVR 軟體中的「**Channel**」參數？作用為何？

**Channel** 是 NVR 中指定頻道號碼的設定參數（請參如下設定畫面），通常是用以指定在多視訊來源的 ONVIF 設備中的某一視訊來源（例如 4-port 的 video server）。若 **Channel** 設定為 **1**，則表示「**第一個頻道**」，**2** 則表示「**第二個頻道**」，依此類推。



用 ONVIF 的術語描述這參數的意義，大約是說從 **GetVideoSources** 所回覆的 video source list 內，只有符合 **Channel** 所指的那個 video source 會被使用，並且由 **GetProfiles** 所回覆的參數中，只有符合前述的 video source 內的參數會被使用到。

### 攝影機的 ONVIF 初始連結取得影像串流程序為何

注意！這是一篇 **ONVIF 實作線索** 的文章，[請先閱讀此處說明](#)。

#### 指令呼叫序列

1. 呼叫 **<device>GetCapabilities** 以取得 ONVIF 各種服務的連接端點（endpoints）。
2. 呼叫 **GetVideoSources** 以取得視訊來源的列表，在 Genius Vision NVR 中，一個「監看頻道」只會使用一個視訊來源，而這個視訊來源則是透過 **Channel** 參數來指定的（[請參閱此處解釋](#)）。若 **Channel** 設定為 **1**，則表示「**第一個頻道**」依此類推。
3. 呼叫 **GetProfiles** 來取得所有的 media profiles 列表。每個視訊來源符合選擇的 profile token，會列在 NVR 的 stream selection 內供使用者選擇。請注意視訊來源不符合 **Channel** 設定的 profile 則不會列出。
4. 呼叫 **<ver20/imaging>GetOptions** 來取得攝影機支援的影像設定參數及範圍。如果這個呼叫失敗，則 NVR 會試圖呼叫 **<ver10/imaging>GetOptions**，這個時候，我們可以說 NVR 已經執行了一個 Imaging service 版本自動偵測的步驟，並把偵測的結果存入 NVR 的一個內部變數稱為 **ImagingVersion**，呼叫成功後，NVR 便會將攝影機所支援的影像設定列在 GUI 上可供使用者選擇。

5. 比對 NVR 內預存的使用者設定的影像設定，以及攝影機目前的影像設定（從 **GetProfiles** 所得知）後，NVR 會決定需要變更攝影機參數的列表，如果 NVR 決定要變更攝影機影像設定，則會呼叫 `<ver20/imaging>SetImagingSettings` 或 `<ver10/imaging>SetImagingSettings`，取決於 **ImagingVersion** 的值。
6. 對於每一個已啟用的 stream，呼叫 **GetVideoEncoderConfigurationOptions**，並將攝影機支援的編碼器設定顯示在 NVR GUI 上。
7. 比對 NVR 內預存的使用者設定的編碼器設定，以及攝影機目前的編碼器設定（從 **GetProfiles** 所得知），NVR 會決定需要變更攝影機參數的列表，如果 NVR 決定要變更攝影機編碼器設定，則會呼叫 **SetVideoEncoderConfiguration**。
8. 呼叫 **GetStreamUri** 以取得 media 的 URI 位址，以進行影音串流，取得回應後，NVR 便會利用 RTSP 進一步連結攝影機。

### 必要指令簡介

從上面的序列中，我們可以知道要正確支援 Genius Vision NVR，則攝影機必須支援一些指令，但這些指令並非每一個都是必要的，必要的指令列舉如下：

- `<device>GetCapabilities` - 必要
- **GetVideoSources** - 必要
- **GetProfiles** - 必要
- **GetStreamUri** - 必要

### 非必要指令簡介

如果攝影機缺乏一些指令支援，則會導致 NVR 必須停用某些功能，但不支援這些指令並不影響攝影機的正常影音串流。

- `<ver20/imaging>GetOptions` 或 `<ver10/imaging>GetOptions` - 非必要。不支援此指令會造成 NVR 提用影像設定的功能。
- `<ver20/imaging>SetImagingSettings` 或 `<ver10/imaging>SetImagingSettings` - optional. Lacking support for this command causes NVR to disable all ImagingOptions functions.
- **GetVideoEncoderConfigurationOptions** - 非必要. Lacking support for this command causes NVR to disable all VideoEncoderOptions functions.
- **SetVideoEncoderConfiguration** - 非必要. Lacking support for this command causes NVR to disable all VideoEncoderOptions functions.

### 如何實作雙重串流 dual-streaming

**注意！這是一篇 ONVIF 實作線索的文章，請先閱讀此處說明。**

#### 為什麼需要雙重串流 dual-streaming?

由於 NVR 顯示影像時仍須透過 CPU 或軟體解碼 H.264 串流，加上近年來 IP 攝影機解析度、畫質及每秒幅數都不斷增加，導致 CPU 解碼效能產生瓶頸，如果要同時顯示多個攝影機（例如 16 個視窗或更多），就有可能產生 CPU 運算力不足的問題，其可能進而影響系統穩定性。

要解決這個問題，最簡單的方式就是採用雙重（或多重）串流，也就是：在視窗數目較少時，用主串流顯示較高的畫質，而在視窗數目較多時，則採用第二串流減輕 CPU 負擔，因此慣例上第一串流是畫質最好的且以 H.264 編碼以利錄影，第二串流以上則以減輕 CPU 負擔為原則（採用較低解析度，或較少 frame-rate），第二串流以上亦可選擇採用 JPEG 以減輕 CPU 解碼負擔。

## 實作

要了解這個部分，請先閱讀以下文章：

- [攝影機的 ONVIF 初始連結取得影像串流程序為何](#)

要實作 ONVIF 雙重串流 dual-streaming，Genius Vision NVR 會先呼叫 **GetProfiles** 以取得支援的 media profile 列表，每一個 profile token 都會列在串流選擇（包含主串流及副串流），因此如果要支援 Genius Vision NVR 中的雙重串流，則 IP 攝影機至少要支援（存在）兩個以上的 media profile。請注意只有匹配選定的視訊來源（也就是 **Channel** 所指定的值，[在這裡有解釋](#)）的 profile token 會被列在 NVR 的串流選擇中。

## 注意事項

為了維持 IP 攝影機的穩定性，Genius Vision NVR 並不會主動呼叫 **CreateProfile** 來新增額外的 Profile（因為攝影機可能本身存在隱藏的限制，這麼做可能會導致攝影機當機）。因此要符合 Genius Vision NVR 的雙重串流，必須攝影機本身就要具備這些 media profile，而非要求 NVR 呼叫 **CreateProfile**。

## 如何實作設定編碼器參數 VideoEncoderOptions (解析度,分辨率,Codec,幀率, FrameRate 等等 etc)

**注意！這是一篇 ONVIF 實作線索的文章，請先閱讀此處說明。**

要了解這個部分，請先閱讀以下文章：

- [攝影機的 ONVIF 初始連結取得影像串流程序為何](#)

若要支援 **VideoEncoderOptions**，攝影機必須實作以下指令：

- **GetProfiles**
- **GetVideoEncoderConfigurationOptions**
- **SetVideoEncoderConfiguration**

## 如何實作設定影像參數 ImagingOptions (亮度, 飽和度, 等)

**注意！這是一篇 ONVIF 實作線索的文章，請先閱讀此處說明。**

要了解這個部分，請先閱讀以下文章：

- [攝影機的 ONVIF 初始連結取得影像串流程序為何](#)

若要支援 **ImagingOptions**，攝影機必須實作以下指令：

- **GetProfiles**，請注意該指令所回傳的[對應](#) Profile 中，其 **Profile/VideoSource/Imaging** 之下的每個值是否存在（包含 **BacklightCompensation, Brightness, ColorSaturation, Contrast, Exposure, Focus, IrCutFilter, Sharpness, WideDynamicRange, WhiteBalance** 等）決定了 NVR 是否能設定該值。
- **<ver20/imaging>GetOptions** 或 **<ver10/imaging>GetOptions**
- **<ver20/imaging>SetImagingSettings** 或 **<ver10/imaging>SetImagingSettings**

## 為什麼需要 ImagingSettings 的資訊

這是因為：在 ONVIF 1.0 中，**ImagingOptions** 的所有項目皆為 mandatory（不可缺省），所以 NVR 軟體無法從 **GetOptionsResponse** 的 **ImagingOptions** 去判斷攝影機支援哪些參數設定。ONVIF 的規範非常完整而龐大，NVR 須盡力避免碰觸攝影機所不支援的部分，以減少不必要的問題因而產生。在 ONVIF

1.0，NVR 軟體要知道攝影機所支援的影像參數設定，唯一的方法是從 **GetVideoSourcesResponse** 裡面的 **ImagingSettings**。因此，在沒有 **ImagingSettings** 的情況，NVR 會認定，攝影機不支援任何影像參數設定。

此邏輯 NVR 在 ONVIF 2.0 規範中亦持續沿用。

觀察 ONVIF 的 Imaging spec 中 1.0 與 2.0 的差異，最重要就在於 **GetOptions** 中每個 element 是否是 mandatory，也就是說：ONVIF 為了這個原因而必須變更 spec 的版本，由此可以判斷，每個攝影機對於參數是否支援的差異，對於 ONVIF 來說是一個非常重要的資訊。

## 如何實作 PTZ 或快速球控制

**注意！這是一篇 ONVIF 實作線索的文章，請先閱讀此處說明。**

### 指令呼叫序列

1. 檢查 NVR 設定參數 **UsePTZ** 內的值，看 PTZ 功能是否有啟用，如果沒有啟用，則 PTZ 的全部功能將停用不起作用。**UsePTZ** 也會指定是否強迫使用 2.0，否則會自動偵測版本 1.0 或 2.0。
2. 呼叫 **<ver20/ptz>GetPresets** 或 **<ver10/ptz>GetPresets**（取決於 **UsePTZ** 的設定值），以取得預設點列表。如果 NVR 沒有設定「強迫使用 2.0」，則 NVR 會透過此指令執行**自動偵測**，也就是：NVR 會先呼叫**<ver10/ptz>GetPresets**，看看攝影機有無正確回應，若有則假設 PTZ 版本為 1.0，若無則假設 PTZ 版本為 2.0。
3. 如果 NVR 收到 PTZ 的滑鼠操作只是，NVR 則會呼叫 **<ver20/ptz>ContinuousMove** 或 **<ver10/ptz>ContinuousMove**（取決於 **UsePTZ** 的設定值）來進行 PTZ 移動操作。
4. 如果 NVR 收到移至預設點(goto preset)的操作，NVR 則會呼叫 **<ver20/ptz>GotoPreset** 或 **<ver10/ptz>GotoPreset**（取決於 **UsePTZ** 的設定值）來進行移至預設點操作。
5. 如果 NVR 收到設定預設點(set preset)的操作，NVR 則會呼叫 **<ver20/ptz>SetPreset** 或 **<ver10/ptz>SetPreset**（取決於 **UsePTZ** 的設定值）來進行設定預設點操作。

## 如何在 ONVIF 的 spec 下實作及支援事件通知 event notification

一般而言，ONVIF 規範內有兩種方式可以支援事件通知：

1. Basic notification
2. Pull-point style notification

由於[此處所述之原因](#)，Genius Vision NVR 並不支援「basic notification」，所以為了通過我們的事件通知驗證，您必須實作「pull-point style notification」。

P.S. 根據 ONVIF 規範，其實存在[第三種方式](#)可以進行事件通知，也就是透過 RTP 的方式把事件嵌入在影音資料內一同串流給 NVC 達到通知的效果，但我們還沒遇過有人有支援，所以自然也無法支援。

### 請參閱

- [如何實作 ONVIF Pull-point style 事件通知 event notification \(ET01\)](#)

## Genius Vision NVR 有支援 basic notification 嗎

不支援。

ONVIF 規範中所定義的 **basic notification**，只能適用於當 IP 攝影機與 NVR 處在同一網段中時才能正確作用，如果 IP 攝影機存在於 Internet 而 NVR 是在 NAT 後面的話，那麼就無法作用，但 NVR 仍可以看到 IP 攝影機的影像，這可能造成使用者產生很大的誤會，而且這個 **callback** 的通知方式，可能需要使用者關閉 NVR 上的防火牆，甚至在系統正式上線後仍需保持防火牆關閉，這樣的要求可能回招致不少爭議。

**Basic notification** 所必須的各種限制及需求，很可能導致現場施工人員及使用單位的嚴重誤解，而且很難在當場就輕易的釐清問題。以實務的觀點，這代表使用 **Basic notification** 會產生數倍的現場支援成本，而這些多出來的成本將會落在參與專案的所有角色身上，包含：IP 攝影機製造商、NVR/VMS 軟體開發商、以及系統整合商。相對來說，若使用「**Pull-point style notification**」則沒有這些問題及疑慮。

因此我們深信：使用「**Pull-point style notification**」對於現場施工人員及客戶來說才是最適當且簡易的方式，如此現場人員不必具備深度網路知識，或必須了解 IP 攝影機及 NVR 的網段資訊等才能正確安裝。

況且如果使用 **basic notification** 就必須要求客戶在系統正式上線後仍需保持防火牆關閉，我們不會建議客人這樣做這種影響系統安全性的操作。

因為以上所述的種種理由，我們也並不打算未來在 **Genius Vision NVR** 中加上 **basic notification** 的支援。

## 如何實作 ONVIF Pull-point style 事件通知 event notification (ET01)

注意！這是一篇 **ONVIF 實作線索** 的文章，[請先閱讀此處說明](#)。

### 指令呼叫序列

1. 這個序列僅適用於 **Event.Template** (請參閱[此處解釋](#)) 設定為 **ET01**。
2. 呼叫 **CreatePullPointSubscription** 取得 HTTP URL endpoint。
3. 利用前步驟所取得的 HTTP URL endpoint，重複呼叫 **PullMessage** 來取得事件通知訊息。事件通知訊息的內容格式已經整理在[此處](#)。(請特別注意 **ET01** 一欄，此欄是從 spec 中整理而來的)

## 如何實作 ONVIF motion detection (位移偵測) 事件 (ET01)

注意！這是一篇 **ONVIF 實作線索** 的文章，[請先閱讀此處說明](#)。

### 指令呼叫序列

1. 這個序列僅適用於 **Event.Template** (請參閱[此處解釋](#)) 設定為 **ET01**。
2. 進行[事件通知序列](#)。

### 事件格式定義

[ONVIF-Imaging-Service-Spec-v221.pdf](#)

### 5.3 Events

The Message structure of these events is given by the following Message Description:

```
<tt:MessageDescription IsProperty="true">
  <tt:Source>
    <tt:SimpleItemDescription Name="Source" Type="tt:ReferenceToken"/>
  </tt:Source>
  <tt>Data>
    <tt:SimpleItemDescription Name="State" Type="xs:boolean"/>
  </tt>Data>
</tt:MessageDescription>
```

The SourceToken points to the source the image is coming from. This is in case of the Analytics or Image Service a VideoSource token and in case of the Recording Service the Recording job token.

#### 5.3.2 MotionAlarm

When a device detects motion (e.g by an Analytics Service) it can inform a client using this event. This event is a basic motion alarm event that should be supported by all devices that support motion detection. If a device has a more complex algorithm running it is free to provide a vendor specific motion alarm event. If the device supports motion detection it should provide the following event.

```
tns1:VideoSource/MotionAlarm
```

#### 通知事件的 XML 範例

```
<wsnt:NotificationMessage>
  <wsnt:Topic Dialect="http://www.onvif.org/ver10/tev/topicExpression/ConcreteSet">
tns1:VideoSource/MotionAlarm
  </wsnt:Topic>
  <wsnt:Message>
    <tt:Message UtcTime="2008-10-10T12:24:57.321Z">
      <tt:Source>
        <tt:SimpleItem Name="Source" Value="[videoSourceToken]" />
      </tt:Source>
      <tt>Data>
        <tt:SimpleItem Name="State" Value="[motionState]" />
      </tt>Data>
    </tt:Message>
  </wsnt:Message>
</wsnt:NotificationMessage>
```

- **[videoSourceToken]**: 必須與 **GetVideoSources** 所回傳的 source token 產生對應。
- **[motionState]**: 如果 motion 正在進行則為「true」，否則為「false」

#### 請參閱

- [How to verify ONVIF motion detection](#)

在 ONVIF 的 event 術語中，什麼是「Property」？什麼是「SetSynchronizationPoint」？要如何實作？

注意！這是一篇 [ONVIF 實作線索的文章](#)，[請先閱讀此處說明](#)。

關於「Property」在規範中的介紹

如 [ONVIF-Core-Specification-v230.pdf](#) 規範中所述，Property 是一種「狀態變化」的簡易表示方式：

#### 9.4 Properties

A Property is a collection of name and value pairs representing a unique and addressable set of data. They are uniquely identified by the combination of their Topic, Source and Key values and are packaged like ordinary events. A Property also contains an additional flag, stating whether it is newly created, has changed or has been deleted.

When a client subscribes to a topic representing a certain property, the device shall provide notifications informing the client of all objects with the requested property, which are alive at the time of the subscription. An client *can* also request the values of all currently alive properties the client has subscribed to at any time by asking for a synchronization point (see section 9.6).

The property interface is defined in this standard in order to group all property related events together and to present uniformly to clients. It is recommended to use the property interface wherever applicable. Section 9.5 explains the structure of events and properties in detail.

Property 操作

根據規範，每一個 Property 都可以有三種操作，分別是：**Initialized** (初始化), **Deleted** (刪除), 以及 **Changed** (改變):

```
<xs:simpleType name="PropertyOperationType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Initialized"/>
    <xs:enumeration value="Deleted"/>
    <xs:enumeration value="Changed"/>
  </xs:restriction>
</xs:simpleType>
```

The PropertyOperation shall be present when the notification relates to a property. The operation mode "Initialized" shall be used to inform a client about the creation of a property. The operation mode "Deleted" shall be used when a synchronization point has been requested.

什麼是 Synchronization Point (同步點)

根據規範：

## 9.6 Synchronization Point

Properties, introduced in section 9.4, inform a client about property creation, changes and deletion in a uniform way. When a client wants to synchronize its properties with the properties of the device, it can request a synchronization point which repeats the current status of all properties to which a client has subscribed. The PropertyOperation of all produced notifications is set to "Initialized" (see Section 9.5). The Synchronization Point is requested directly from the SubscriptionManager which was returned in either the SubscriptionResponse or in the CreatePullPointSubscriptionResponse. The property update is transmitted via the notification transportation of the notification interface. The following operation shall be provided by all Subscription Manager Endpoints:

請特別注意紅框中的字：Synchronzation Point(同步點) 在呼叫 **CreatePullPointSubscription** 的時候就自動產生了，這意思是說：NVC（也就是用戶端）並不需要主動呼叫 **SetSynchronizationPoint**。

NVC (用戶端) 需要主動呼叫 **SetSynchronizationPoint** 嗎  
不需要。

請參閱上節解釋，這一點在規範中定義得很明確，無模糊地帶。

究竟什麼是 Synchronization Point(同步點)? 我還是不明白

ONVIF 規範習慣將簡單的概念用專業的術語來表達，因此有時不容易明白。作法簡單來說，就是在 NVT 每次收到 **CreatePullPointSubscription** 要求時，必須假設 NVC 完全不知道目前的 Property 狀態，此時 NVT 必須將所有的 Property 狀態以 **PropertyOperation="Initialized"** 的方式全部告知 NVT (透過 **NotificationMessage** 的形式)，如此一來 NVC 才能夠知道所有 Property 的初始狀態，未來有變更時（以 **PropertyOperation="Changed"**）NVC 才能正確知道是從何種狀態改變成何種狀態。

舉例來說，如果某一個 digital input 目前是在 **ON** 的狀態，當 NVT 收到 **CreatePullPointSubscription** 要求的時候，他必須立即送出 **ON** 的 Property 狀態事件給 client (並表示操作類型 **PropertyOperation="Initialized"**)，如此一來 client 就不會因為缺乏資訊而誤認或假設這個 digital input 初始狀態為 **OFF**，並且在未來狀態改變時能夠正確執行動作。

如果 NVC 不需要呼叫 **SetSynchronizationPoint** 指令，那麼這指令存在的意義為何？  
根據規範：

Table 97: SetSynchronizationPoint command

SetSynchronizationPoint		Access Class: READ_MEDIA
Message name	Description	
SetSynchronizationPoint-Request	<i>This message is empty.</i>	
SetSynchronizationPoint-Response	<i>This message is empty.</i>	
Fault codes	Description	
	<i>No command specific faults!</i>	

When a client uses the **notification streaming interface**, the client should use the SetSynchronizationPoint operation defined in the ONVIF Media Service Specification.

**SetSynchronizationPoint** 指令其實是保留給 **notification streaming interface** (這是什麼?) 事件通知介面用的。Genius Vision NVR 只會使用 Pull-Point style event notification，因此這個指令永遠不會被 Genius Vision NVR 呼叫。

### 何謂 notification streaming interface 事件通知?

根據 ONVIF spec ([ONVIF-Core-Specification-v230.pdf](#)),

#### 9.3 Notification Streaming Interface

Section "Metadata Configuration" of the ONVIF Media Service Specification describes the creation, deletion and modification of metadata configurations. Certain metadata configurations can contain multiple subscriptions whose structure is the same as that for a notification subscription. When a metadata configuration containing subscriptions has been assigned to a profile, a client uses that profile to get an RTP stream that includes the configured notifications as metadata. The notification streaming via RTP shall be implemented by an ONVIF compliant device that supports the ONVIF Media service.

The [WS-BaseNotification] defines the element `wsnt:NotificationMessage` to pack the Message Payload, the Topic and the ProducerReference. The structure of this message is the same as that for direct notification requests (the format is described in Section 9.5). Multiple instances of the `wsnt:NotificationMessage` elements can be placed within a metadata document introduced in the Real-time Viewing section.

There is no explicit SubscriptionReference with streaming notifications. Therefore, the `wsnt:NotificationMessage` shall not contain the SubscriptionReference element.

此種特殊的通知介面只能配合與 RTP/RTSP 一併使用，在此模式下，NVT 將通知訊息嵌入在特殊的 RTP 封包中以達到事件通知的目的，Genius Vision NVR 目前不支援這種事件通知介面。

### 如何實作 ONVIF 的警報輸入 Digital Input (ET01)

注意！這是一篇 **ONVIF 實作線索** 的文章，[請先閱讀此處說明](#)。

## 指令呼叫序列

對於 Genius Vision NVR 如何使用警報輸入 digital input 的程序可整理為如下序列：

1. 這個序列僅適用於 **Event.Template** (請參閱[此處解釋](#)) 設定為 **ET01**.
2. 呼叫<device>**GetDigitalInputs** 以取得可用的警報輸入 digital input 列表。如果失敗 NVR 會嘗試呼叫 <ver10/deviceio>**GetDigitalInputs**.
3. 進行[事件通知序列](#)。

取得警報輸入 digital input 的狀態變化

規範中事件定義

[ONVIF-DeviceIo-Service-Spec-v221.pdf](#)

### 5.13.1 DigitalInput State Change

A device that signals support for digital inputs in its capabilities shall provide the following event whenever one of its input state changes:

Topic: tns1:Device/Trigger/DigitalInput

```
<tt:MessageDescription IsProperty="true">
  <tt:Source>
    <tt:SimpleItemDescription Name="InputToken" Type="tt:ReferenceToken"/>
  </tt:Source>
  <tt>Data>
    <tt:SimpleItemDescription Name="LogicalState" Type="xs:boolean"/>
  </tt>Data>
</tt:MessageDescription>
```

Digital Input LogicalState can be either set at "true" to represent the circuit in the closed state or set at "false" to represent the circuit in the open state.

## 通知事件的 XML 範例

```
<wsnt:NotificationMessage>
  <wsnt:Topic Dialect="http://www.onvif.org/ver10/tev/topicExpression/ConcreteSet">
tns1:Device/Trigger/DigitalInput
  </wsnt:Topic>
  <wsnt:Message>
    <tt:Message UtcTime="2008-10-10T12:24:57.321Z">
      <tt:Source>
        <tt:SimpleItem Name="InputToken" Value="[digitalInputToken]" />
      </tt:Source>
      <tt>Data>
        <tt:SimpleItem Name="LogicalState" Value="[digitalInputState]" />
      </tt>Data>
    </tt:Message>
  </wsnt:Message>
</wsnt:NotificationMessage>
```

- **[digitalInputToken]**: 必須與<device>**GetDigitalInputs** 或 r <ver10/deviceio>**GetDigitalInputs** 所回傳的 token 產生對應。
- **[digitalInputState]**: 如果是 ON 則為「 true 」，OFF 則為「 false 」

See also

- [How to setup & verify ONVIF DIO \(Digital Input/Relay Output\)](#)

## 如何實作 ONVIF 繼電器輸出控制 Relay Output (ET01)

注意！這是一篇 [ONVIF 實作線索的文章](#)，請先閱讀此處說明。

### 指令呼叫序列

對於 Genius Vision NVR 如何使用繼電器輸出控制 Relay Output 的程序可整理為如下序列：

1. 這個序列僅適用於 **Event.Template** (請參閱[此處解釋](#)) 設定為 **ET01**.
2. 呼叫 `<device>GetRelayOutputs` 以取得 relay 的數量及對應的各個 token，如果這個呼叫失敗，那麼 NVR 會嘗試呼叫 `<ver10/deviceio>GetRelayOutputs`. 這個時候我們可以說一個自動版本偵測已經執行，並且將偵測結果存在一個內部變數 **RelayUseDevIO** 中。
3. 進行[事件通知序列](#)。
4. 如果使用者操作 relay 成為 ON 或 OFF，NVR 就會依據 **RelayUseDevIO** 的值呼叫 `<device>SetRelayOutputState` 或 `<ver10/deviceio>SetRelayOutputState`。

取得繼電器輸出控制 relay output 的狀態變化

規範中事件定義

[ONVIF-DeviceIo-Service-Spec-v221.pdf](#)

#### 5.13.2 Relay Output Trigger

event

A device that signals RelayOutputs in its capabilities should provide the Trigger event whenever its relay inputs change. An ONVIF compliant device shall use the following topic and message format:

Topic: `tns1:Device/Trigger/Relay`

```
<tt:MessageDescription IsProperty="true">
  <tt:Source>
    <tt:SimpleItemDescription Name="RelayToken" Type="tt:ReferenceToken"/>
  </tt:Source>
  <tt>Data>
    <tt:SimpleItemDescription Name="LogicalState" Type="tt:RelayLogicalState"/>
  </tt>Data>
</tt:MessageDescription>
```

### 通知事件的 XML 範例

Relay 的狀態改變是必須使用事件通知的方式（因為其狀態有可能透過其他方式改變）：

```
<wsnt:NotificationMessage>
  <wsnt:Topic Dialect="http://www.onvif.org/ver10/tev/topicExpression/ConcreteSet">
tns1:Device/Trigger/Relay
  </wsnt:Topic>
  <wsnt:Message>
    <tt:Message UtcTime="2008-10-10T12:24:57.321Z">
      <tt:Source>
        <tt:SimpleItem Name="RelayToken" Value="[relayToken]" />
      </tt:Source>
      <tt>Data>
        <tt:SimpleItem Name="LogicalState" Value="[relayLogicalState]" />
      </tt>Data>
    </tt:Message>
  </wsnt:Message>
</wsnt:NotificationMessage>
```

```
</tt:Message>
</wsnt:Message>
</wsnt:NotificationMessage>
```

- **[relayToken]**: 必須與<device>GetRelayOutputs 或 <ver10/deviceio>GetRelayOutputs（取決於 **RelayUseDevIO**）所回傳的 token 產生對應。
- **[relayLogicalState]**: **active** 或 **inactive**，取決於繼電器的實際狀態。

### 控制繼電器輸出 relay output 改變狀態

- 攝影機必須支援<device>GetRelayOutputs 或 <ver10/deviceio>GetRelayOutputs 以表示其支援繼電器輸出控制 relay output。
- 攝影機必須實作<device>SetRelayOutputState 或 <ver10/deviceio>SetRelayOutputState 以接受控制繼電器輸出 relay output 改變狀態。
- 以上兩個所需支援的指令之 spec 版本必須一致。

### 請參閱

- [How to setup & verify ONVIF DIO \(Digital Input/Relay Output\)](#)

### 什麼是 InitialTerminationTime 爭議？如何實作

注意！這是一篇 **ONVIF 實作線索** 的文章，[請先閱讀此處說明](#)。

### 規範中的線索

關於 **CreatePullPointSubscription** 指令中的 **InitialTerminationTime** 參數之意義，ONVIF 規範中寫得不是很明確，但可以如下解釋之：

- 如果 **InitialTerminationTime** 是用 relative time(相對時間)定義的，那麼關於 TerminationTime 的計算方面：camera 必須在每次 client 呼叫 **PullMessageRequest** 完成後自動延長 **InitialTerminationTime** 所指定的值。

### 可能的爭議

我們必須了解：TerminationTime 存在的意義在於將不再存活的 Subscription 刪除以節省資源，有些工程師會將 **InitialTerminationTime** 解釋成「絕對停止時間且無法延長」，但請仔細想想：一個 **Subscription** 若有無法延長的絕對停止時間，在實務上並沒有意義，ONVIF 規範中已經告訴我們「**PullMessageRequest** 的一個重要功能在於表示 keep-alive 訊號」，也就是告知 NVT 該 Subscription 仍然存活，因此上節的解釋應該是很容易推理的。

我們在寫程式的時候，除了逐字解釋規範外，還要用腦袋想想：這麼做有沒有意義。

### NVR 支援狀態

- 根據[此處所述的原則](#)，為了解決「有些工程師將 **InitialTerminationTime** 解釋成：絕對停止時間且無法延長」的問題，我們在後來的版本中加入了 **ET02** 參數以解決這個爭議。

### 技術細節

以上的解釋 spec 中並沒有明確定義，但可以從 spec 中的 XML 通話範例中推導而出：(以下節錄自 [ONVIF-Core-Specification-v230.pdf](#) 中的內容並加上紅色旁白):

### 9.12.3 CreatePullPointSubscription

A client can subscribe to specific notifications with the information from the TopicProperties. The following XML example shows the subscription for notifications produced by the Rule Engine of the device. The client reacts only to notifications that reference VideoAnalyticsConfiguration "2" and VideoSourceConfiguration "1". The Subscription has a timeout of one minute. If the subscription is not explicitly renewed or messages are not pulled regularly, it will be terminated automatically after this time.

```
<?xml version="1.0" encoding="UTF-8"?> Pay attention to ambiguous description
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing"
  xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
  xmlns:tet="http://www.onvif.org/ver10/events/wsd1"
  xmlns:tns1="http://www.onvif.org/ver10/topics">
  <SOAP-ENV:Header>
    <wsa:Action>
      http://www.onvif.org/ver10/events/wsd1/EventPortType/CreatePullPointSubscriptionRequest
    </wsa:Action>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <tet:CreatePullPointSubscription>
      <tet:Filter>
        <wsnt:TopicExpression
          Dialect="http://www.onvif.org/ver10/tev/topicExpression/ConcreteSet">
          tns1:RuleEngine//.
        </wsnt:TopicExpression>
        <wsnt:MessageContent
          Dialect="http://www.onvif.org/ver10/tev/messageContentFilter/ItemFilter">
          boolean(//tt:SimpleItem[@Name="VideoAnalyticsConfigurationToken"
            and @Value="2"] ) and
          boolean(//tt:SimpleItem[@Name="VideoSourceConfigurationToken"
            and @Value="1"] )
        </wsnt:MessageContent>
      </tet:Filter>
      <tet:InitialTerminationTime>
        PT1M
      </tet:InitialTerminationTime>
    </tet:CreatePullPointSubscription>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

#### 9.12.4 CreatePullPointSubscriptionResponse

When the device accepts the Subscription, it returns the `http://160.10.64.10/Subscription?Idx=0` URI which represents the Endpoint of this Subscription. Additionally, the client is informed about the `CurrentTime` of the device and the `TerminationTime` of the created Subscription.

```
<?xml version="1.0" encoding="UTF-8"?>
  <SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://www.w3.org/2005/08/addressing"
    xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
    xmlns:tet="http://www.onvif.org/ver10/events/wsd1">
    <SOAP-ENV:Header>
      <wsa:Action>
http://www.onvif.org/ver10/events/wsd1/EventPortType/CreatePullPointSubscription
Response
      </wsa:Action>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
      <tet:CreatePullPointSubscriptionResponse>
        <tet:SubscriptionReference>
          <wsa:Address>
            http://160.10.64.10/Subscription?Idx=0
          </wsa:Address>
        </tet:SubscriptionReference>
        <wsnt:CurrentTime>
          2008-10-09T13:52:59
        </wsnt:CurrentTime>
        <wsnt:TerminationTime>
          2008-10-09T13:53:59
        </wsnt:TerminationTime>
      </tet:CreatePullPointSubscriptionResponse>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

Termination time is computed initially.

#### 9.12.5 PullMessagesRequest

The client sends a `PullMessagesRequest` to the Endpoint given in the `CreatePullPointSubscriptionResponse` to get Notifications corresponding to a certain Subscription. The following sample request contains a `Timeout` of five (5) seconds and limits the total number of messages in the response to two (2).

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing"
  xmlns:tet="http://www.onvif.org/ver10/events/wsd1" >
  <SOAP-ENV:Header>
```

```
<wsa:Action>
http://www.onvif.org/ver10/events/wsdl/PullPointSubscription/PullMessagesRequest
</wsa:Action>
  <wsa:To>http://160.10.64.10/Subscription?Idx=0</wsa:To>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
  <tet:PullMessages>
    <tet:Timeout>
      PT5S
    </tet:Timeout>
    <tet:MessageLimit>
      2
    </tet:MessageLimit>
  </tet:PullMessages>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

## 9.12.6 PullMessagesResponse

The following PullMessageResponse contains two notifications which match the subscription. The Response informs the client that two objects have crossed lines corresponding to rules "MyImportantFence1" and "MyImportantFence2".

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing"
  xmlns:wstop="http://docs.oasis-open.org/wsn/t-1"
  xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
  xmlns:tet="http://www.onvif.org/ver10/events/wsdl"
  xmlns:tns1="http://www.onvif.org/ver10/topics"
  xmlns:tt="http://www.onvif.org/ver10/schema">
  <SOAP-ENV:Header>
    <wsa:Action>
      http://www.onvif.org/ver10/events/wsdl/PullPointSubscription/PullMessagesResponse
    </wsa:Action>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <tet:PullMessagesResponse>
      <tet:CurrentTime>
        2008-10-10T12:24:58
      </tet:CurrentTime>
      <tet:TerminationTime>
        2008-10-10T12:25:58
      </tet:TerminationTime>
      <wsnt:NotificationMessage>
        <wsnt:Topic
          Dialect="http://www.onvif.org/ver10/tev/topicExpression/ConcreteSet">
          tns1:RuleEngine/LineDetector/Crossed
        </wsnt:Topic>
        <wsnt:Message>
          <tt:Message UtcTime="2008-10-10T12:24:57.321Z">
            <tt:Source>
              <tt:SimpleItem Name="VideoSourceConfigurationToken"
                Value="1"/>
              <tt:SimpleItem Name="VideoAnalyticsConfigurationToken"
                Value="2"/>
              <tt:SimpleItem Value="MyImportantFence1" Name="Rule"/>
            </tt:Source>
            <tt:Data>
              <tt:SimpleItem Name="ObjectId" Value="15" />
            </tt:Data>
          </tt:Message>
        </wsnt:Message>
      </wsnt:NotificationMessage>
      <wsnt:NotificationMessage>
        <wsnt:Topic
```

Termination time is updated, not by client

```
    tns1:RuleEngine/LineDetector/Crossed
  </wsnt:Topic>
  <wsnt:Message>
    <tt:Message UtcTime="2008-10-10T12:24:57.789Z">
      <tt:Source>
        <tt:SimpleItem Name="VideoSourceConfigurationToken"
          Value="1"/>
        <tt:SimpleItem Name="VideoAnalyticsConfigurationToken"
          Value="2"/>
        <tt:SimpleItem Value="MyImportantFence2" Name="Rule"/>
      </tt:Source>
      <tt:Data>
        <tt:SimpleItem Name="ObjectId" Value="19"/>
      </tt:Data>
    </tt:Message>
  </wsnt:Message>
</wsnt:NotificationMessage>
</tet:PullMessagesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

