International Journal of Control and Automation
Vol. 8, No. 2 (2015), pp. 163-172
http://dx.doi.org/10.14257/ijca.2015.8.2.17

Design and Implementation of an ONVIF Proxy Server

Hansol Ji[1], Min Choi[2] and Namgi Kim[1*]

[1] *Department of Computer Science, Kyonggi University*
{jhs572, ngkim}@kgu.ac.kr
[2] *School of Info. and Comm. Eng., Chungbuk National University*
mchoi@cbnu.ac.kr

Abstract

The Open Network Video Interface Forum (ONVIF) has created a standard protocol for improving compatibility between the protocols employed by heterogeneous network cameras. The aim of this study was to design and implement a proxy server to reduce the management cost of network video transmitters (NVTs) by reducing the load of NVTs as well as the network load of NVTs.

Keywords: *ONVIF, NVT, Proxy server*

1. Introduction

At present, numerous closed circuit televisions (CCTVs) are employed in various areas and the development of networks has allowed anyone to easily manage CCTVs. Users can obtain information by accessing each CCTV, which should provide the services requested by users. However, the presence of a high number of CCTV users generates a large load and thus a greater bandwidth is required for networks that connect CCTVs. In this study, we aimed to resolve this problem by moving the CCTV information transmission capacity to a proxy server, thereby allowing users to be provided with services by receiving information from the proxy server using a reverse proxy server. In addition, we aimed to improve the versatility of this system by implementing the proxy server in a manner that satisfies Open Network Video Interface Forum (ONVIF) standards [1].

2. Related Work

The CERN proxy [2] is a web proxy that uses three methods to determine the expiration time of objects, as follows.

1. An expires header is employed.
2. CacheDefaultExpiry is used to calculate the Time-to-Live (TTL) for objects.
3. If there is no last-modified field, CacheDefaultExpiry specifies the default TTL of the cached object.

The Wessels proxy server does not use the fork system for calls, but uses multithreading. The performance of the Wessels proxy server and the CERN proxy server were compared in [3], which showed that the Wessels proxy server performed slightly better because it did not incur high costs for overhead context switching and the creation of threads during multithreading. The Wessels proxy server comprises three sub-modules: cache_mgr, web-proxy, and reached.

The Squid proxy [4] performs non-blocking I/O to avoid overheads when creating a process after receiving a new request, where a single process handles each request. A three-phase structure is cached in disks. Sixteen sections are used to cache the files into

* Corresponding author: Namgi Kim
This paper is a revised and expanded version of a paper entitled "A Study on ONVIF Proxy Server" presented at SIP 2014, Haikou, China, and December 21th 2014.

ISSN: 2005-4297 IJCA

256 subdirectories and fingerprinting is employed to map the URL and the names of objects.

3. ONVIF Proxy Server Design

3.1 Overview

Table 1. Terminaology

Abbreviation	Full name
DDM	Device Discovery Module
PMM	Process Management Module
PPM	Proxy Process Module
RCM	RTSP Client Module
RSM	RTSP Server Module
LDD	Legacy Device Discovery
ODD	ONVIF Device Discovery

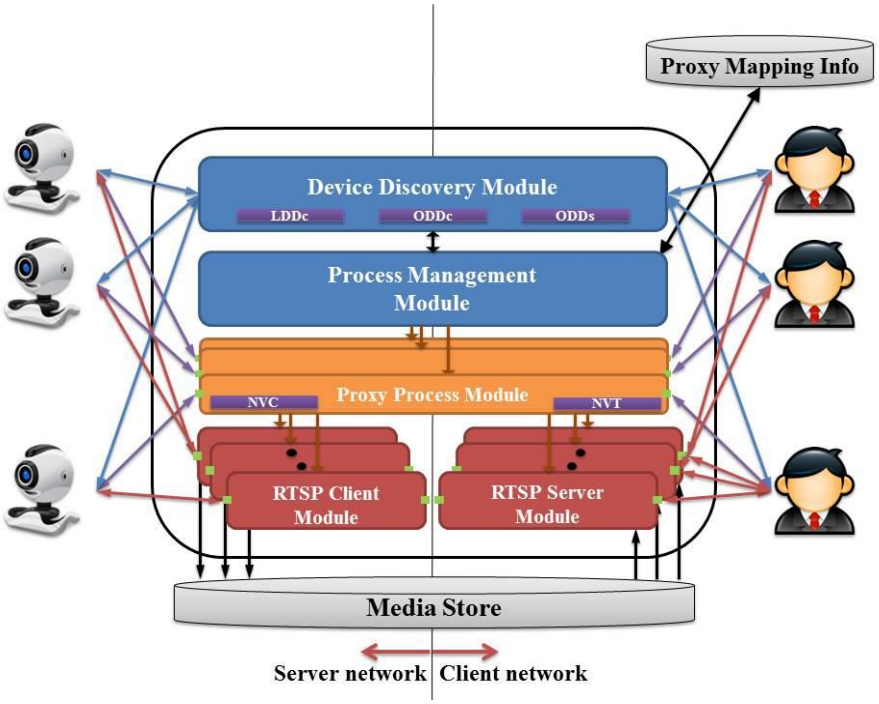

Figure 1. Overall Structure

Table 1 show the abbreviations used in this study. Figure 1 shows the overall structure of the proxy server. The left-hand side shows the server network where the network video transmitters (NVTs) are located. The right-hand side shows the client network where the network video clients (NVCs) are located. The device discovery module (DDM) finds NVTs in the server network side and sends discovery messages to NVCs on the client network side. The DDM was implemented based on WS-Discovery. The process management module (PMM) manages various proxy process modules (PPMs) and it maps information between NVTs and their corresponding PPMs. Therefore, if an access request arrives from NVCs, it delivers its IP and port number, which is mapped to each NVT. The mapping information includes each NVT's IP, the client network side's IP:Port for the PPM that corresponds to the respective NVT, and the last renewal time. One PPM is produced for each NVT and each PPM appears to be an NVT to clients. In addition, because the PPMs should appear to be clients to the NVTs and as NVTs to clients, they

should be capable of playing the roles of both NVCs and NVTs. If PPMs are created, they are connected with NVTs. If the connection occurs without any problems, the RTSP client module (RCM) and the RTSP server module (RSM) are produced. The RCM and the RSM relay the streaming between NVTs and NVCs. The RCM stores image information received from NVTs in media storage. The RSM obtains image information stored in the media storage and transmits it to users.

3.2 Procedures

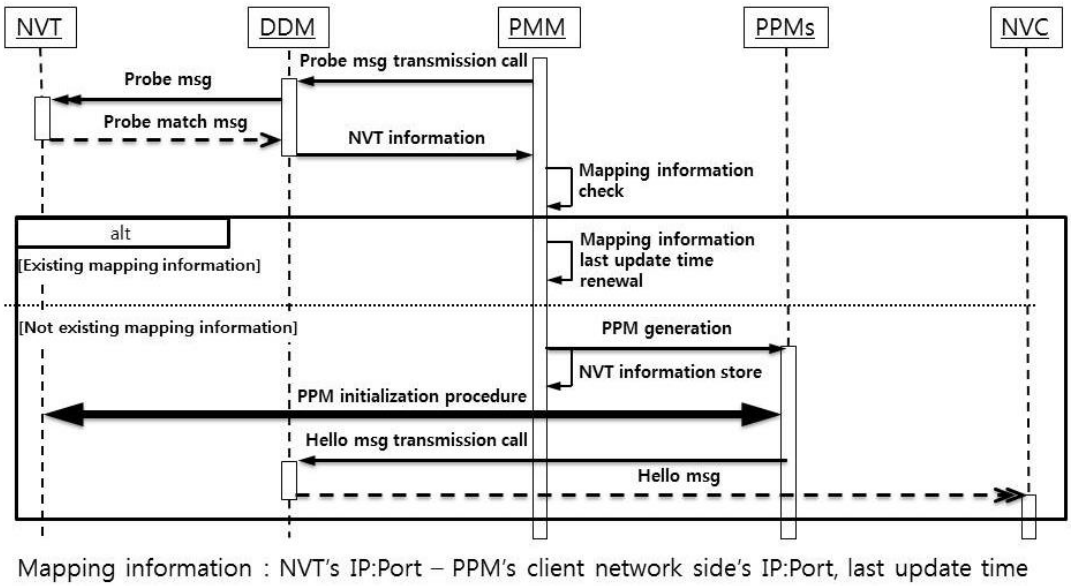

Figure 2. Periodic Proxy Server Procedure

Figure 2 shows the procedure performed by the proxy server every hour. The PPM requests that the DDM transmits a Probe message to the server network side. The Probe message is transmitted by the DDM, but not directly by the PMM because the standard WS-Discovery uses a port number of 3702. If the DDM transmits the Probe message via multicast, NVTs that have received the message send the probe match message to the DDM. If the DDM delivers information about NVTs in the probe match message, the PMM checks its own mapping information and the information for the respective NVTs. If information about the respective NVTs is present in the mapping information, only the renewal time of the mapping information is initialized. If information about the respective NVTs is not present in the mapping information, it is considered to be related to a new NVT. Therefore, a PPM is created to connect with the new NVT and new mapping information is produced. If a PPM is created, the connection procedure between the NVT and the PPM is carried out in compliance with the ONVIF standards, and the access to the new NVT is notified by transmitting a hello message to the client network side via multicast.

Figure 3 shows the procedure when a PPM is created due to the access of a new NVT. The PPM accesses and connects with the NVT in compliance with ONVIF standards. First, if the PPM transmits GetCapabilitiesRequest to the NVT, the NVT transmits GetCapabilitiesResponse, thereby sending its own capabilities to the PPM. The PPM receives the end point of the media service in the NVT's capabilities and it sends GetProfilesRequest to the media service. The NVT sends its own profiles by transmitting GetProfilesResponse to the PPM and it then receives an RTSP URI by requesting a URI

based on the profiles. If the URI is received, the RCM and RSM are created, and the RCM receives images by accessing the NVT's RTSP URI.

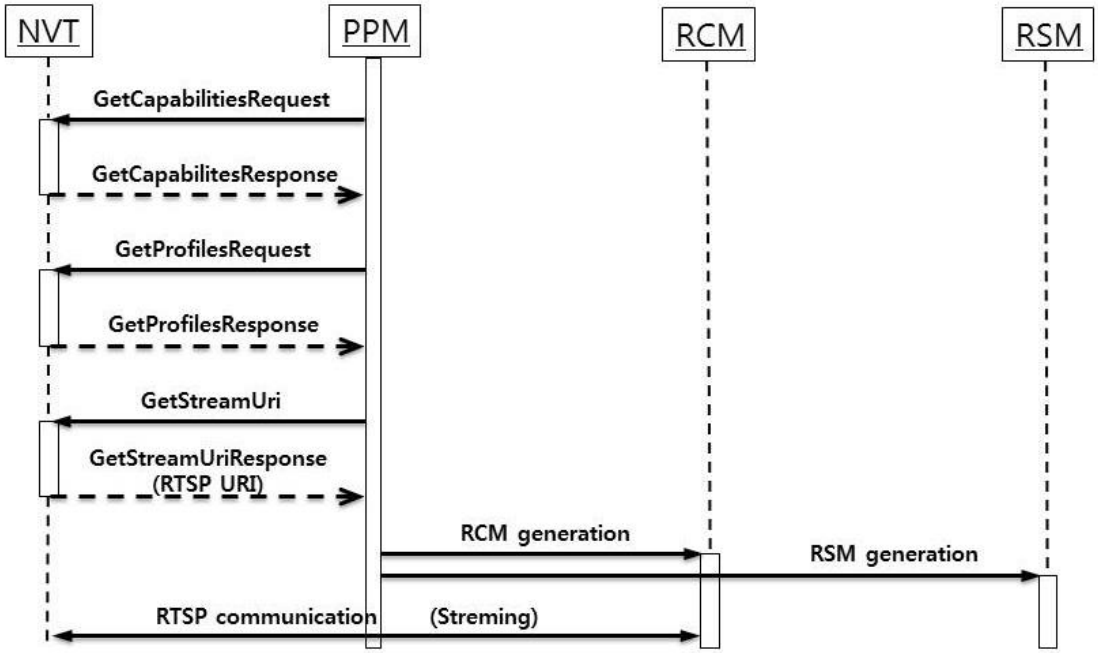

Figure 3. PPM Initialization Procedure

Figure 4 shows the procedure when the NVC transmits a Probe message. If the DDM receives the Probe message, the DDM informs the PMM that the NVC intends to have access and the PMM checks its own mapping information, before creating all of the NVT's information in the form of a Probe match message and sending a request for transmission to the DMM.

Figure 5 shows the procedure performed when the NVC has a new access. If the NVC has a new access, an accessible NVT is searched for using the probe message and the information related to the PPMs is obtained. Based on the information obtained, GetCapabilitiesRequest is transmitted to the PPM that needs to be accessed and the PPM that receives the message delivers the respective message to the connected NVT. The header is modified to disguise the message sent to the NVT so it resembles one sent by the proxy server, *i.e.*, it includes the proxy server create and the message's security component is sent. After receiving GetCapabilitiesResponse from the NVT, the end points of the contents are changed to the proxy server's address and the remaining connection procedure is performed.

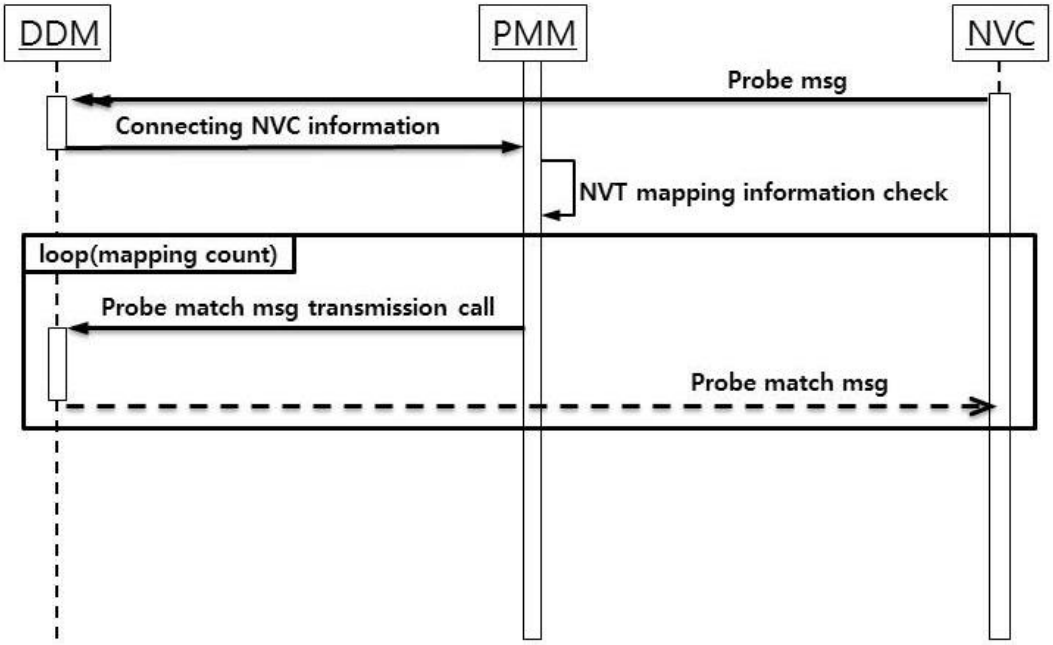

Figure 4. NVC Discovery Procedure

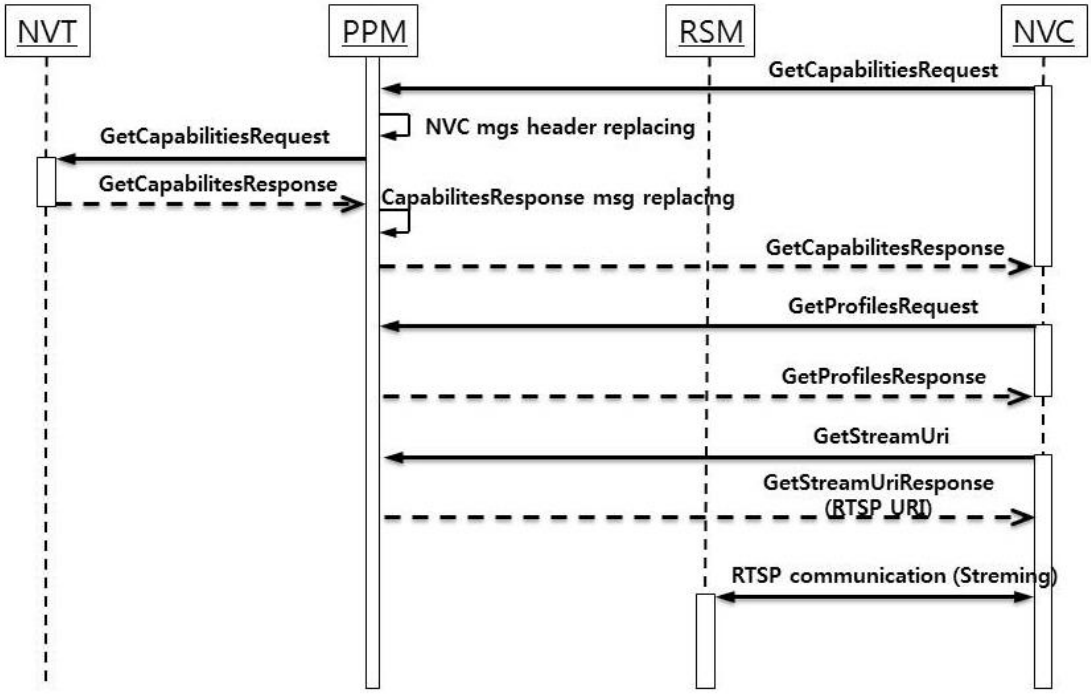

Figure 5. NVC Initialization Procedure

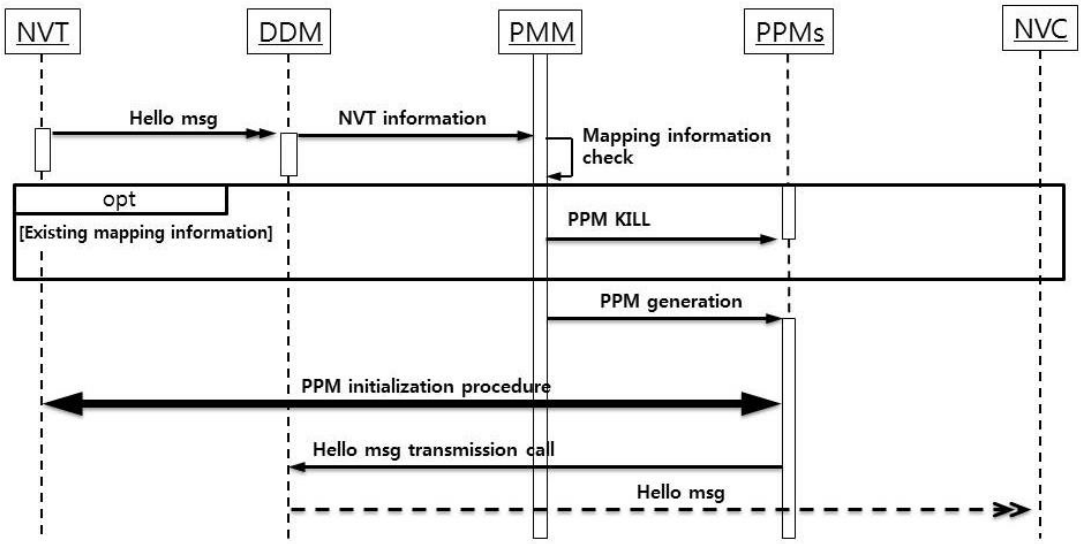

Figure 6. NVT Initalization Procedure

Figure 6 shows the procedure when the new NVT accesses and sends a Hello message. The newly accessed NVT transmits a Hello message to the network via multicast. If the DDM transmits the Hello message, it notifies the newly accessed NVT's information to the PMM. The PMM identifies whether this information overlaps with the NVT's information by checking its own current mapping information. If the respective NVT's information already exists within the mapping information, it is considered to be the sudden termination and reconnection of an existing NVT. Therefore, the PPM that has been connected with the respective NVT is terminated forcibly. If there is no information or the PPM is generated again after terminating the existing PPM, the PPM initialization procedure is performed. Next, the NVT's access to the client network side is notified by sending a Hello message.

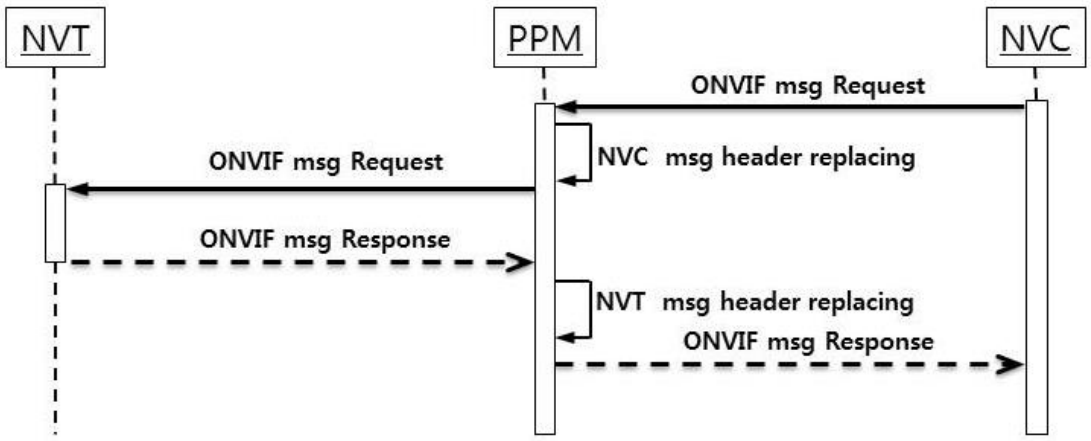

Figure 7. ONVIF Message Handling Procedure

Figure 7 shows the procedure when the PPM handles an ONVIF message after it is sent from the NVC. The PPM that receives the ONVIF massage modifies the respective message's header and forwards the output to the NVT. If the NVT transmits the response and the PPM receives it, the PPM then modifies the response message's header again and forwards the output to the NVC.

Table 2. Inter Process Communication Format

Type	Transmission condition	Transmission location	Receive location	Data
1	Hello message arrival	DDM	PMM	NVT info
2	Probe match message arrival	DDM	PMM	NVT info
3	Probe message transmission request	PMM	DDM	PPM's port number list
4	Probe match message transmission request	PMM	DDM	PPM's port number list
5	Hello message transmission request	PPM	DDM	Port number
6	PPM generation	PMM	PPM	NVT IP to be connected
7	PPM generation	PPM	PMM	Allocated port number
8	RCM, RSM generation	PPM	RCM, RSM	RTSP URI

Table 2 shows the messages transmitted between the different processes. In Type 1, the DDM receives a Hello message from the NVT and delivers the received information to the PM. The PMM is provided with the NVT's information to notify that the Hello message has been received. In Type 2, after the DDM receives the Probe match message from the NVT and delivers the received information to the PMM, it provides the PMM with the NVT's information and a notification that the probe match message has been received. Mutually different messages are received and the same NVT's information is delivered to the PMM, but different behaviors can be identified after the receipt of the messages, which should be identified. Type 3 is the message type sent by the PMM to the DDM to transmit the message to the server network side, where a list of the port numbers of PPMs is delivered. Type 4 is the message sent by the PPM to the DDM to transmit the Probe match message after receiving the probe message from the NVC, where a list of the port numbers of PPMs is delivered. Type 5 is a message for requesting a Hello message, which is sent by the PPM mapped to the respective NVT when it needs to notify the new NVT's access to the client network side. Types 6 and 7 are messages that the PMM and the PPM need to deliver to each other when a PPM is created. The PMM delivers the IP of the NVT that should be connected to the PPM. The PPM delivers the port number assigned to itself to the PMM. Type 8 is the message delivered by the PPM when the RCM and RSM are created. The RTSP URI that the RCM needs to access is delivered.

4. Implementation Results

In this study, tests were performed to verify the validity of the implementation results. Figures 8 and 9 show SNC-CH120 [5] and WV-SC384 [6], which was used in this test, and they satisfied ONVIF. The PC used in the client role sent ONVIF messages with the verification tool provided by ONVIF. The proxy server received these messages and sent them to the respective CCTV. If a response was received, the server sent it to the client. In the tests, we confirmed whether the ONVIF verification tool sent and received messages without problems. Figure 10 shows the test results, which demonstrate that the GetCapabilitiesResponse was received normally in response to the GetCapabilitiesRequest.

In this study, we designed and implemented a proxy server that satisfies ONVIF standards. We also performed tests that verified the implementation and its actions were confirmed by the tests. In future research, we aim to demonstrate the practical feasibility of the proxy server by applying it in actual environments.

Figure 8. SNC-CH120(Sony)

Figure 9. WV-SC384(Panasonic)

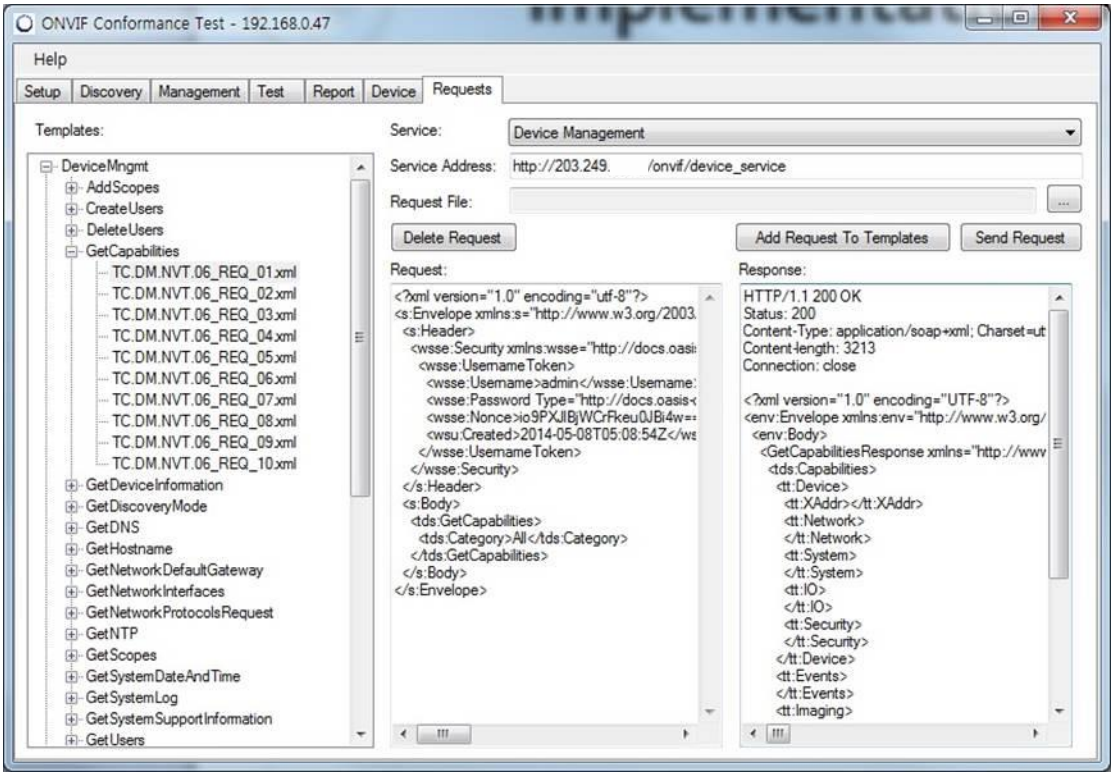

Figure 10. ONVIF Conformance Test 1.02

4. Acknowledgment

This research was supported by a grant (13SCIPA01) from Smart Civil Infrastructure Research Program funded by Ministry of Land, Infrastructure and Transport(MOLIT) of Korea government and Korea Agency for Infrastructure Technology Advancement(KAIA).

References

[1]. "ONVIF", www.onvif.org.
[2]. "CERN proxy", https://espace.cern.ch/ webservices-help/CERNLibraryProxy/Pages/default.aspx.
[3]. A. Dingle and T. Part, "web Cache Coherenc", Fifth Internetional World Wide Web Conference, Paris, **(1996).**
[4]. "Squid proxy", http://www.squid-cache.org.
[5]. "SNC-CH120", http://www.pro.sony.eu/pro/lang/en/eu/product/video-security-ip-cameras-fixed/snc-ch120/overview/.
[6]. "WV-SC384", http://www.bhphotovideo.com/c/product/850451-REG/Panasonic_WV_SC384_WV_SC384_HD_Dome_Network.html/c/product/#inpage:IN+STOCK?gclid=CjgKEAjwuMmdBRDljdfi2_qQpxkSJADDCRwsvnzHht2J6p86vaiQx1uyK6HlIxMFR-j1jh78jVMyO_D_BwE.

Authors

Hansol Ji, he received the B.S. degree in Computer Science from the Kyonggi University, Korea, in 2013. He is currently M.S. candidate in Computer Science from Kyonggi University. his research interests include wireless systems, sensor networks.

Min Choi received the B.S. degree in Computer Science from Kwangwoon University, Korea, in 2001, and M.S. and Ph.D. degrees in Computer Science from Korea Advanced Institute of Science and Technology (KAIST) in 2003 and 2009, respectively. From 2008 to 2010, he worked for Samsung Electronics as a Senior Engineer. Since 2011 he has been a faculty member of Department of Information and Communication of Chungbuk National University. His current research interests include embedded system, computer architecture, and mobile cloud.

Namgi Kim, received the B.S. degree in Computer Science from Sogang University, Korea, in 1997, and the M.S. degree and the Ph.D. degree in Computer Science from KAIST in 2000 and 2005, respectively. From 2005 to 2007, he was a research member of the Samsung Electronics. Since 2007, he has been a faculty of the Kyonggi University. His research interests include sensor system, wireless system, and mobile communication.