XML-based Configuration Management for IP Network Devices

DPNM Lab
Hyoun-Mi Choi
2003. 12. 13
siwa@postech.ac.kr
Content

- Introduction
- Overview of Netconf
- Problems & Proposed Solutions
- Architecture of XML-based Configuration Management System (XCMS)
- Implementation
- Testing and Evaluation
- Conclusions and Future Work
Introduction

- As the Internet continues to grow
  - IP networks are complex and composed of diverse network devices
  - the network management of IP networks are becoming more and more difficult

- Problem in SNMP-based Network Management
  - Limited Management Information Modeling: SMI MIB
    - Not support structured data types, objects, or methods
    - Not support to present the relationships among managed objects
  - Limited Management Protocol: SNMP
    - Not support to provide various operations to improve management functionalities (Get/Set/Trap/Getbulk)
    - Not support to retrieve a large volume of data using SNMP over UDP
Introduction (2)

• SNMP Problem for Configuration Management
  – Need to process bulk data (no `setbulk` operation)
  – Provide static addressing method to map every object to the OID number
  – Provide complex relationships between managed objects

• Revolutionary approach to solve the exiting problems
  – Management Information Modeling considering relationships
  – Management Protocol using SOAP
  – Addressing Method: XPath

=>Propose Architecture of XML-based Configuration Management System (XCMS)
Overview of Netconf

- A working group in IETF which was formed in May 2003
- Standardize XML-based management protocol message to provide interoperability
  - Based on a remote procedure call (RPC)
  - Define various RPC operations to improve the management functionality
- Use connection-oriented protocol for transport considering bulk data processing
Netconf Management Protocol

Layer

- Content
- Operations
- RPC
- Transport

Example

- Configuration data
- <get-config>, <edit-config>
- <rpc>, <rpc-reply>
- BEEP, SSH, SOAP over HTTP
Defined XML Tags of Netconf

- **Operation Layer**
  - Base RPC Operations
    - `<get-config>`, `<edit-config>`
    - `<copy-config>`, `<delete-config>`
    - `<lock>`, `<kill-session>`
  - Additional RPC Operations

- **RPC Layer**
  - Non operations
    - `<rpc>`, `<rpc-reply>`
  - RPC Operations
    - `<rpc-abort>`, `<rpc-abort-reply>`
    - `<rpc-progress>`
Transport Layer

- **Session**: the connections between the manager and the agent
- **Channels**
  - **Operation Channel** *(MUST)*
    - carry a series of RPCs in the NETCONF operations layer
  - **Management Channel** *(MAY)*
    - carry information of managing the NETCONF session
  - **Notification Channel** *(MAY)*
    - carry notifications from the agent to the manager
## Application Protocol Binding

<table>
<thead>
<tr>
<th></th>
<th>SSH</th>
<th>BEEP</th>
<th>SOAP over HTTP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management Channel</td>
<td>Not</td>
<td>support</td>
<td>Not</td>
</tr>
<tr>
<td>Operation Channel</td>
<td>support</td>
<td>support</td>
<td>support</td>
</tr>
<tr>
<td>Notification Channel</td>
<td>Not</td>
<td>support</td>
<td>Not</td>
</tr>
<tr>
<td>RPC Interface</td>
<td>Not</td>
<td>Not</td>
<td>Support</td>
</tr>
<tr>
<td>Independent-transport</td>
<td>Not</td>
<td>Not</td>
<td>Support</td>
</tr>
<tr>
<td>Note</td>
<td>Server/Client</td>
<td>peer-to-peer</td>
<td>Server/Client</td>
</tr>
</tbody>
</table>
<edit-config> Message

```xml
<rpc message-id="107" xmlns="http://ietf.org/xmlconf/1.0/base">
  <edit-config>
    <target>
      <running/>
    </target>
    <config xmlns="http://example.com/schema/1.2/config" xmlns:xc="http://ietf.org/xmlconf/1.0/base">
      <interface>
        <name>Ethernet0/0</name>
        <mtu>1500</mtu>
        <address xc:operation="replace">
          <name>1.2.3.4</name>
          <mask>255.0.0.0</mask>
        </address>
      </interface>
    </config>
  </edit-config>
</rpc>
```

Transport: SSH, BEEP, SOAP over HTTP

XCMS
(XML-based Configuration Management System)
Problem in Netconf protocol

- Provide unclear layer between Operation and Content layers because of no addressing method (\texttt{<edit-config>})
- Not support to describe RPC operations to improve management functionalities
Problem in Netconf over SOAP/HTTP (2)

- Not provide multiple channels
- Not call mandatory RPC operations by SOAP RPC interface
- Not define WSDL of RPC operations to guarantee interoperability between SOAP RPC operations

Management protocol using SOAP RPC needs WSDL of RPC operations
Our Solutions and Suggestions

- To overcome SNMP problems
  - Suggest **management information modeling** considering relationships using XML Schema
  - Suggest **SOAP-based management protocol**

- To overcome Netconf problems
  - Use **XPath** as a addressing method
  - Suggest only XPath expressions to need in configuration management on agent aspect
  - Define RPC operations using **WSDL**
  - Include **HTTP Server/Client** to support notification channel
# Management Protocol: SOAP

<table>
<thead>
<tr>
<th>NETCONF Protocol Level</th>
<th>SOAP RPC Message Level</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content</td>
<td>Parameter</td>
<td></td>
</tr>
<tr>
<td>Operation</td>
<td>SOAP RPC Operation</td>
<td></td>
</tr>
<tr>
<td>&lt;get-config&gt;,</td>
<td>&lt;get-config&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;edit-config&gt;,</td>
<td>&lt;edit-config-replace&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;lock&gt;,</td>
<td>&lt;edit-config-merge&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;kill-session&gt;,</td>
<td>&lt;edit-config-delete&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;copy-config&gt;,</td>
<td>&lt;copy-config&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;delete-config&gt;,</td>
<td>&lt;delete-config&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;lock&gt;,</td>
<td>&lt;lock&gt;,</td>
<td></td>
</tr>
<tr>
<td>&lt;kill-session&gt;,</td>
<td>&lt;kill-session&gt;,</td>
<td></td>
</tr>
<tr>
<td>etc.</td>
<td>&lt;download&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;upload&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;reboot&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;shutdown&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;rpc-progress&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;rpc-abort&gt;</td>
<td></td>
</tr>
<tr>
<td>RPC</td>
<td>&lt;rpc&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;rpc-progress&gt;,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;rpc-abort&gt;,</td>
<td></td>
</tr>
<tr>
<td>Transport</td>
<td>HTTP</td>
<td>HTTP</td>
</tr>
</tbody>
</table>

//SOAP RPC Operation
<SOAP-ENV:Body id="50">
<edit-config-replace>
<target>
<running/>
</target>
<xpath />
<wanset>
<config>
<wanset>
<ip>211.171.1.1</ip>
</wanset>
<gatewayip>
211.171.1.2
</gatewayip>
<wanset>
<config>
<edit-config-replace>
</SOAP-ENV:Body>
Using SOAP RPC Interface

Manager

- SOAP Server
- SOAP over HTTP
- SOAP client

Agent

- SOAP Server
- SOAP over HTTP
- SOAP client

Base & Additional WSDL

Notify WSDL

Notify
Base
Additional

Base & Additional WSDL

Notify WSDL
Using SOAP RPC Interface

Manager

SOAP Server

Notify

SOAP Server

SOAP over HTTP

SOAP Client

SOAP over HTTP

Agent

SOAP Client

SOAP Server

Notify

Notify

Base & Additional WSDL

Notify WSDL

Notify WSDL
Addressing Method: XPath

//ip[@name='primary'] | //ip[@name='second']

- **Node-set** is an unordered collection of nodes without duplicates
- **Operators** are used for selecting the nodes within a specific range
Management Information Modeling

```xml
<IPGW>
  <basic>
    <lan>
      <ip>192.168.10.1</ip>
      <networkmask>255.255.255.0</networkmask>
    </lan>
    <wan>
      <auto ref='//dpcp/set'>disable</auto>
      <wanset>
        <ip>211.171.108.16</ip>
        <networkmask>255.255.255.192</networkmask>
        <gatewayip>211.171.108.1</gatewayip>
      </wanset>
    </wan>
    <dns>
      <ip name='primary'>192.168.1.1</ip>
      <ip name='second'>192.168.1.2</ip>
      <ip name='third'></ip>
    </dns>
  </basic>
  <dhcp>
    <set ref='//wan/auto'>disable</set>
    <dhcpset>
      <startingip>155</startingip>
      <usernumber>30</usernumber>
    </dhcpset>
  </dhcp>
</IPGW>
```
Architecture of XCMS

Manager

- Web Browser
- Web Interfaces
- RPC Operations
- Local Operations
- XML Parser

Agent

- Web Browser
- Web Interfaces
- RPC Operations
- Local Operations
- XML Parser

Repository

- XMLDB
- File
  - Log
  - XLS
- RPC Operation
  - WSDL
  - Configurations

Repository

- XMLDB
- File
  - Log
  - XLS
- RPC Operation
  - WSDL
  - Configurations
Implementation

**Manager**
- Management Operations
  - Web Interfaces
    - XSLT
    - Xalan
  - RPC Operations
  - Local Operations
    - XML Parser
      - Xerces
- Repository
  - XMLDB
  - Xindice
  - Log
  - XLS
  - Configurations

**Agent**
- Management Operations
- RPC Operations
- Local Operations
  - XML Parser
    - Libxml
- Repository
  - Log
  - Configurations

**Interactions**
- SOAP over HTTP
  - Request
  - Response

**Tools**
- SOAP
- XML Parser
- Libxml
- Xerces
- Axis
- gSoap
- Xalan
- Xindice

**Documents**
- WSDL
- Log
- Configurations
Performance Test Environment

XCMS Manager
(Pentium III
800MHz, CPU 256MB)

100 Mbps

XCMS Agent

SNMP Manager
(Pentium III
800MHz, CPU 256MB)

100 Mbps

SNMP Agent
Performance Evaluation (1)

- SNMP(Get/GetBulk) vs. XCMS(get-config)
- Network traffic overhead
  - Transferred Message Size (Request & Response)
- System processing overhead
  - Packet Numbers from request to response
  - Response time from request to response
Request

Number of MIB objects

Transferred Message Size (Byte)

SNMP (get)
SNMP (getbulk)
XCMS (get)
Response

- SNMP(get)
- SNMP(getbulk)
- XCMS(get)

Transferred Message Size (Byte) vs Number of MIB objects
Packet

Number of MIB objects

Packet

SNMP (get)
SNMP (getbulk)
XCMS (get)
Response Time

![Graph showing response time vs number of MIB objects for different methods: SNMP(get), SNMP(getbulk), and XCMS(get).]

- **Response Time (sec)**: The y-axis represents the response time in seconds.
- **Number of MIB objects**: The x-axis represents the number of MIB objects.
- The graph compares the response time for SNMP(get), SNMP(getbulk), and XCMS(get) methods as the number of MIB objects increases.

---

2004-03-04

XCMS

(XML-based Configuration Management System)
## Comparison and Summary

<table>
<thead>
<tr>
<th></th>
<th>SNMP</th>
<th>Netconf over SOAP</th>
<th>XCMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Bulk data</strong></td>
<td>Limit (Get)</td>
<td>No limit (Get&amp;Set)</td>
<td>No limit (Get&amp;Set)</td>
</tr>
<tr>
<td><strong>Extend or Modify operations</strong></td>
<td>Difficult</td>
<td>Easy</td>
<td>Easiest Using WSDL</td>
</tr>
<tr>
<td><strong>Notification</strong></td>
<td>Support (trap)</td>
<td>Not</td>
<td>Support</td>
</tr>
<tr>
<td><strong>Relationships of objects</strong></td>
<td>Not</td>
<td>Not (Out of Content)</td>
<td>Support</td>
</tr>
<tr>
<td><strong>Addressing Method</strong></td>
<td>Absolute path</td>
<td>Not</td>
<td>Relative path</td>
</tr>
</tbody>
</table>
Conclusions

- Proposed Architecture of XML-based Configuration Management System (XCMS)
- Optimized management protocol for SOAP binding
- Provides the ease of implementation using XML technologies
- Guarantees the interoperability using SOAP RPC operations defined by WSDL
Future Work

- Further to optimize XCMS agent in an embedded system
- Plan to extend XCMS to Web Services
Published Papers List


Thank you!!
Problem in Netconf over SOAP

XML-based Configuration Management System (XCMS)

```
<soapenv:body>
  <rpc message-id="107"
        xmlns="http://ietf.org/xmlconf/1.0/base">
    <edit-config>
      <target>
        <running/>
      </target>
      <config xmlns="http://example.com/schema/1.2/config"
               xmlns:xc="http://ietf.org/xmlconf/1.0/base">
        <interface>
          <name>Ethernet0/0</name>
          <mtu>1500</mtu>
          <address xc:operation="replace">
            <name>1.2.3.4</name>
            <mask>255.0.0.0</mask>
          </address>
        </interface>
      </config>
    </edit-config>
  </rpc>
</soapenv:body>
```
Conclusions (1)

- Management protocol
  - SOAP message is over connection-oriented protocols. This is useful for processing bulk data.
  - SOAP provides the Remote Procedure Call (RPC) interface. This easily calls new additional operations.
  - SOAP-based protocol supports to easily modify the message format.
  - SOAP tools generate the WSDL files which contain the definitions of the RPC operations.
SNMP vs. XML-based CM

- **Management Information Modeling**
  - XML defines various data types using XML Schema
  - XML presents the dependency of nodes

- **Addressing Method**
  - XPath is composed of the names of the nodes
  - XPath supports a relative path because of accessing with name of the node
SNMP vs. XML-based CM

- Management Protocol

**<Request PDU>**

| PDU Type | Request-id | 0 | 0 | Variable bindings |

**<Response PDU>**

| PDU Type | Request-id | Error-status | Error-index | Variable bindings |

**<SNMP Message Formats>**

```xml
<SOAP-ENV:Body id="107">
  <get-config>
    <target>
      <running>
        </target>
    </xpath>
    //admin
  </xpath>
  <get-config>
  </SOAP-ENV:Body>
```

**<SOAP-based Message Formats>**

```xml
<SOAP-ENV:Body>
  <get-config>
    <config>
      ....................... (XML data)
    </config>
  </SOAP-ENV:Body>
```