A Lightweight C2 Service Invocation Method Based on HTTP Proxy

Dr. Heng Wang

National Key Lab of Science and Technology on C4ISR
Nanjing, Jiangsu Province, P.R.China
Email: puchengew@yahoo.com.cn
Outline

- Background & Motivation
- Service-Oriented C2 Software Architecture
- Our Proposed Method
- Experiments
Background & Motivation
SOA (Service-Oriented Architecture) is an architectural approach and methodology that builds on the concept of services.
Web services (SOAP, WSDL, UDDI, …) are a set of open standards that will lead to widespread adoption of SOAs and serve as the basis for a new generation of service oriented development.
SOA and Web Services technologies have been increasingly applied to military fields.

- GIG/NCES
- FORCENET, FCS, JBI
- Command and Control domain: NECC
- ...

Background & Motivation
In service-oriented C2 systems, software and capability of C2 systems often are wrapped **web services** (or other style services) so as to implement C2 function and information exchanging through service invocation.
Background & Motivation

Service Discovery (e.g., WSDL, UDDI)

Service Provider

Service Consumer

Service invocaiton

Bind/Invoke (HTTP, SOAP)

XML Document

SOAP

HTTP

Service Run-time Environment (e.g., .NET, Websphere, ...)

APP Server + Service Run-time Environment
Background & Motivation

Problems

- XML efficiency
- Security of HTTP
- Unauthorized access
- Interpolated
- Intercepted

Problems

- Unauthorized access
- Interpolated
- Intercepted

XML efficiency

Security of HTTP

Service

Service Consumer

Service Provider

Service invocation

Bind/Invoke (HTTP, SOAP)

Discover (WSDL, UDDI)

Publish (WSDL, UDDI)

Service Description

Service Description

SOAP
Our work

- Proposed a service-oriented C2 software architecture
- Proposed a lightweight service invocation method based on HTTP proxy
Service-Oriented C2 Software Architecture
Service-Oriented C2 Software Architecture (SO-C2SA)

- "service" as the granularity of the C2 software architecture
- unified description mechanism
- summarize elements of domain software and the correlation between the elements
- achieve service combination and orchestration based on process approach
- the mission capability packages (MCP) for the carrier of service capability
Service-Oriented C2 Software Architecture

C2 MCP
- Force Projection Services
- Air/Space Operations Services
- Joint Fires & Maneuvers Services
- Situational Awareness Services
- Intel Services
- Executive Summary Services

C2 Service Infrastructure
- C2 Support Services
  - Process Mgt Services
  - Alert Services
  - Resource Mgt Services
  - User Mgt Services
  - Visualization Services
- C2 Common Services
  - Locator Services
  - Report Mgt Services
  - Entity Mgt Services
  - Fusion Services
  - Oceanography Services

Global Information Grid
- CES Infrastructure
  - Security Services
  - Mediation Services
  - M2M Messaging Services
  - Service Discovery
- Computing Infrastructure
- Communications Infrastructure

SO-C2SA
Mission-oriented C2 architecture application pattern
Our Proposed Method
Our basic idea

- based on the principle of **HTTP proxy**
- In HTTP proxy, as an application-level gateway, the proxy server plays the role of the bridge between the client and the server.
Our basic idea --- **Prerequisite**

- **LAN environment is secure**, and allowed to use some open or standard transport protocols such as HTTP
- use military transport protocols for messaging in **WAN environment**.
Our Proposed Method

Our basic idea

- in LAN, standard Web Services are used to messaging, i.e., SOAP via HTTP
- deploy HTTP proxy software at the boundary of the LAN to intercept HTTP packets
- The SOAP message in the HTTP packets is delivered in the WAN via the military transport protocol
- At the receiver end, the same HTTP proxy is used to convert and restore standard invocation way (i.e., SOAP via HTTP) in LAN
Our Proposed Method

LAN

Service Consumer
Request
Response

SOAP Process

Service Run-time Environment
(e.g., .NET, Websphere, ...)

WAN

XML Document

SOAP

HTTP

Service Provider
Request
Response

APP Server +
Service Run-time Environment

LAN

XML Document

SOAP

HTTP

HTTP Proxy

Service Consumer
Request
Response

SOAP Process

HTTP

Service Run-time Environment
(e.g., .NET, Websphere, ...)

XML Document

SOAP

HTTP

Military Transport Protocol

Packet containing SOAP content
Experiments
Experiments

- Experiments environment

Diagram:
- LAN 192.168.11.x
- HTTP proxy client
- HTTP proxy server
- App server
- WAN
- C2 computing service
- 192.168.22.x
- Client
- Server

Network setup:
- Client connects to HTTP proxy client.
- HTTP proxy client connects to HTTP proxy server.
- HTTP proxy server connects to App server.
- App server sends data to C2 computing service.
Experiments

- Compared methods
  - C2 service invocation not using HTTP proxy (NO_PROXY)
  - Based on HTTP proxy, messaging between two HTTP proxies is implemented by using Socket (SOCK_PROXY)
  - Based on HTTP proxy, messaging between two HTTP proxies is implemented by using military transport protocol (MIL_PROXY)
Experiment 1: the total time of invocation
Experiment 2: the time in MIL_PROXY mode
**Experiment 3: the quantity of data**

<table>
<thead>
<tr>
<th></th>
<th>NO_PROXY</th>
<th>SOCK_PROXY</th>
<th>MIL_PROXY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Client</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The size of request packet (1st)</td>
<td>407</td>
<td>407</td>
<td>474</td>
</tr>
<tr>
<td>The size of request packet (2nd)</td>
<td>305</td>
<td>305</td>
<td>352</td>
</tr>
<tr>
<td><strong>APP Server</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The size of request packet (1st)</td>
<td>25</td>
<td>25</td>
<td>66</td>
</tr>
<tr>
<td>The size of request packet (2nd)</td>
<td>572</td>
<td>572</td>
<td>617</td>
</tr>
</tbody>
</table>
Our proposed method

- Simple
- Effective
- Feasible
- Meet the security and real-time requirements in military environment
Thank You!