CAROLINA TECHNOLOGY CONSULTANTS

Tech Brief

July 2009 *Architectural Issues in IT and Data Communications*

Topic

SOAP

Prepared By

Tyler R. Johnson

SOAP

----

* An extensible protocol for exchanging messages.
* Increasingly the protocol of choice for web services architectures.
* Can be bound over any transport protocol.

Summary

SOAP is an extensible and generic protocol for exchanging messages in distributed networks. Built on XML, SOAP is able to carry information adapted to a wide variety of applications and can be transported over heterogeneous networks.

Discussion

SOAP provides a basic message type that is transported among nodes on the network. Several important concepts characterize SOAP:

SOAP Messages

SOAP messages are XML information sets (i.e. structured XML documents) comprising the fundamental data structure of SOAP. SOAP messages consist of an *envelope*, a *header* with *header blocks*, a *body* and *fault* elements. A *message exchange pattern* (*MEP*) defines the ways in which particular messages can be structured.

SOAP Nodes

A SOAP node is an element on the network that communicates SOAP messages. Nodes can take three forms: an *initial SOAP sender*, a *SOAP intermediary*, or an *ultimate SOAP receiver*. SOAP nodes MUST process SOAP messages. Nodes are identified by their URIs.

SOAP Features

A *SOAP feature* is a valid extension of SOAP created to achieve a specific behavior. *SOAP applications* conform to the SOAP model and process SOAP messages.

SOAP Binding

A SOAP *binding* is a concrete specification for the communication of SOAP messages over specific protocols, such as TCP, HTTP, etc.

The SOAP processing model is stateless, relying on single messages. It is the responsibility of the SOAP feature (i.e. application) to maintain state and correlate discrete SOAP messages.

SOAP message headers contain routing information that supports the transportation of messages from the initial sender, along zero or more intermediaries, to the ultimate receiver. Intermediaries may read and modify the messages.

SOAP itself does not provide security mechanisms. Security can be implemented in several ways:

1. At the transport layer beneath SOAP (e.g. TLS or SSL).
2. Using a SOAP extension to refine the message content and processing behavior.
3. At the application level to ensure that malicious SOAP applications sending malformed SOAP messages do not gain unsecured access or cause vulnerabilities.

SOAP Part 2 defines an HTTP binding which inherits security attributes from HTTP. Through this binding, SOAP can be carried over HTTPS.

Since SOAP uses alphanumeric character encodings (XML) it does not efficiently transport binary coded data. For this reason, the SOAP Message Transmission Optimization Mechanism (MTOM) has been defined.

Strategy Considerations

Because SOAP can run a variety of network transport protocols, it presents interesting opportunities for flexibility, such via Bluetooth, visible light communication or RSS.

SOAP is increasingly the protocol architecture of choice for web services, and is gaining traction in the embedded systems community as well. REST may be considered as an alternative to SOAP that has less overhead, although providing less structure as well. Because of XML in particular and the design style of web services architectures in general, SOAP applications implementing web services tend to be inefficient in both message size and processing requirements for parsing, although emergent compression schemes appear to be able to create extremely efficient binary XML representations.

For Further Information

1. W3C Recommendation SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), M. Gudgin, M. Hadley, N. Mendelsohn, J. Moreau, H. Nielsen, A. Karmarkar, Y. Lafon, April 2007
2. WC3 Recommendation SOAP Version 1.2 Part 2: Adjuncts (Second Edition), M. Gudgin, M. Hadley, N. Mendelsohn, J. Moreau, H. Frystyk Nielsen, A. Karmarkar, Y. Lafon, April 2007

3. WC3 Recommendation SOAP Message Transmission Optimization Mechanism, M. Gudgin, N. Medelsohn, M. Nottingham, H. Ruellan, January 2005

4. W3C Recommendation Web Services Addressing 1.0 – Core, M. Gudgin, M. Hadley, T. Rogers May 2006

5. W3C Recommendation Web Services Addressing 1.0 - SOAP Binding, T. Rogers, M. Gudgin, M. Gudgin, M. Hadley, M. Hadley, T. Rogers, May 2006