Master's Thesis Project Specification: Verification of the Session Management Protocol

Student: Karl Palmskog (palmskog@kth.se)
Supervisor at EAB: Yuri Ismailov (yuri.ismailov@ericsson.com)
Supervisor at KTH: Mads Dam (mfd@kth.se)
Examiner: Johan Håstad (johanh@nada.kth.se)

Date: May 22, 2006

Background

During the past two decades, significant attention has been paid by the research community to the problem of providing mobility management for network hosts. This has resulted in a number of solutions for reliably allowing changes in network attachment for devices running the TCP/IP protocol suite. Following the rise of wireless communication and of multi-access capable terminals requiring dynamic reconfiguration, there is an increasing demand for such network mobility. However, obstacles for Internet mobility include the increasing the degree of network heterogeneity, which requires handovers between e.g. IPv4 and IPv6 networks, and the change in focus from device mobility to session mobility.

Moreover, changes in user behaviour have led to a demand for a more intuitive interface between devices and network-enabled applications. Having session control functions such as "start", "stop", "pause", "rewind" and "fast forward" can be greatly beneficial for users who regularly experience interruptions in network access — due to e.g. dynamic behaviour or outages.

One approach to providing delay-, disconnection- and mobility-tolerant communication is to exploit and enhance features which were specified for the session layer in the network protocol stack. Work at Ericsson Research has recently resulted in a description and a proof-of-concept implementation of the Session Management Protocol (SMP), which aims to provide such communication reliability. The thesis project work concerns the formal verification of this protocol.

Motivation

Communication protocols are one the few types of programs which regularly undergo formal analysis and verification. This has to do with both the difficulty of designing a correct protocol and the devastating consequences of errors in a widely implemented specification. However, communication protocols, as non-terminating concurrent programs, are not amenable to ordinary program analysis in terms of input and output. Instead, a mathematical framework for concurrent, interactive systems must be used.

Currently, the application of formal methods for concurrent programs is not a trivial task and requires knowledge of many fields of computer science, such as automata theory, mathematical logic, proof theory and programming language semantics. But at the same time, there have been progress towards increasing the usability of formal methods and related software tools — practitioners need no longer be mathematicians.

Project description

Input for the project

- Specification of the Session Management Protocol
- Specification of the TCP/IP protocol suite

Project objectives

The Session Management Protocol aims to provide session control for any type of network session, but the current focus is on TCP sessions. The challenge in using TCP at the transport layer lies in the fact that SMP and TCP are both stateful protocols which interact during a session. It is necessary that SMP does not break the TCP semantics of providing reliable, in-order data transfer between communicating hosts. Furthermore, TCP and SMP have different timeout mechanisms for session control, which all need to work as expected.

In light of the complexity of the TCP and SMP state machines and the protocol interactions, it is clear that a formal evaluation of SMP, which focuses on internal consistency and TCP feature interaction, is required. Potential problems include unexpected protocol behaviour leading to ambiguous states or deadlock. Hence, project work includes formally describing and verifying SMP, and thoroughly documenting verification results.

Project tasks

- Understand SMP and TCP behaviour and interaction from specification.
- Provide and motivate choice of formal methods for the verification of SMP.
- Provide a formal description of SMP.
- Provide a formal description of SMP/TCP interaction.
- Apply formal methods for verification of SMP and obtain results. This may be an iterative process, where verification results prompt changes in the formal system description.
- Summarize results, and if necessary, suggest changes to SMP specification.
- Produce a report on the work done.

Project scope

The thesis work concerns the *correctness* of the SMP protocol, not its performance and security. Verification may lead to suggestions of changes in the SMP protocol specification, but actually making these changes to the proof-of-concept implementation and testing them is not necessary. Similarly, verification of components in the framework in which SMP resides, such as the name resolution mechanism, is outside the project scope.

Methodology

Literature on formal methods often suggest the following general methodology for the verification of communication protocols and other software systems:

1. Choose a mathematical framework capable of representing concurrent systems.
2. Formulate a tool-independent formal description and specification of the protocol.
3. Select a software tool incorporating verification algorithms and adapt the model and specification to its format.
4. Run the tool using the formal protocol description and specification.
5. Evaluate and summarize verification results.

This methodology will be used for the thesis project. The only equipment needed for completing the steps are computers for running the software tools.

Analysis of related work

Work relevant to the task of verifying SMP can be divided into categories as below. The reading part of the project will be examined by a thesis chapter deliverable.

Network mobility approaches and solutions

Includes reading of papers and reports relating to the network protocol mobility problem situation, and to the session-layer mobility approach. Also, analysis of the master's theses specifying SMP and its framework is required.

TCP/IP and protocol verification

Includes finding and reading material on the formal analysis of protocols. The focus is on protocols in the TCP/IP suite, especially TCP. Of special interest are formal descriptions of the TCP state machine and their properties.

Theories for formal description of concurrent systems

Includes evaluating the available mathematical frameworks for describing and analyzing concurrent, communicating systems. Some examples are the Calculus of Communicating Systems (CCS), the π-calculus, and Communicating Sequential Processes (CSP). Moreover, such a study cannot neglect relevant parts of automata theory and programming language semantics.

Formal specification of correctness properties

Includes evaluating frameworks for formal specification, of which modal and temporal logics are an important part. A limiting factor is the availability of effective model checking algorithms, which must be investigated. Another area of interest is the bisimulation theory of process algebra.

The feature-interaction problem

Includes finding and reading material on the feature-interaction problem of software systems, especially as it pertains to automata and communication protocols. This is of interest when analyzing the behaviour of the SMP and TCP state machines and how they interact.

Time schedule

1. **Analysis of related work**

   - network mobility approaches and solutions
   - TCP/IP and protocol verification
   - theories for formal descriptions of concurrent systems
   - formal specification of correctness properties
   - the feature-interaction problem

   Estimated time: 5 weeks
   Deliverable: thesis Table of Contents with section abstracts, incorporating a literature study report
   Deliverable due: June 5, 2006

2. **Choice, and motivation, of formal methods**

   - selection of mathematical framework
   - selection of modeling language
   - model-checking algorithms
   - specification formalisms
   - software tools

   Estimated time: 1 week
   Deliverable: thesis chapter section on choice of formal methods and decision procedure
   Deliverable due: June 12, 2006

3. **Formal system descriptions**

   - formal SMP description
   - formal description of SMP/TCP interaction
   - adaption of formal descriptions to selected software tool

   Estimated time: 3 weeks
   Deliverable: system descriptions
   Deliverable due: July 3, 2006

4. **System specifications**

   - formulation of formal protocol correctness properties
   - adaption of property specifications to selected software tool

   Estimated time: 2 weeks
   Deliverable: system specifications
   Deliverable due: July 24, 2006

5. **Verification of systems according to specification**

    - application of software tool
    - reiteration of system descriptions and correctness specification
    - error detection and documentation

    Estimated time: 4 weeks
    Deliverable: report on verification results
    Deliverable due: September 1, 2006

6. **Summary of results**

    - evaluation of protocol errors and possible specification changes
    - writing thesis

    Estimated time: 5 weeks
    Deliverable: thesis draft
    Deliverable due: October 2, 2006

Specification approval

Name	**Date**
Karl Palmskog	
Yuri Ismailov	
Mads Dam	
Johan Håstad	