



**MODULAR PROGRAMME
ASSESSED COURSEWORK SPECIFICATION**

Module Details:

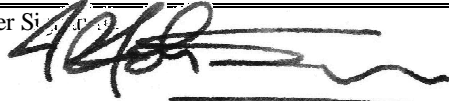
Module Code: UFSEJ6-10-3	Module Title: Advanced Distributed Systems	
Module Leader: (please print) Ian Johnson		Module Tutors: (continued)
Module Tutors: Ian Anderson		
Assignment Element Number: (e.g. CW1) CW1		Weighting: (% of the module's assessment) 50
		Total Assignment Time: (hours) 20

Dates:

Assignment issued to student Date: 22/02/05	Assignment to be returned to student Date:
Submission Place: THE POST BOXES IN N BLOCK FOYERS BOXES WILL BE MARKED WITH THE MODULE CODE TAKE CARE TO POST THE WORK IN THE CORRECT BOX (boxes are open for submission from the Monday before the submission date)	Submission Date: 21/04/03
	Submission Time: 10:00 am

Deliverables:

As per attached specification

Module Leader Signatures: 

You may work in pairs, or individually for this assignment. Groups larger than two are not acceptable.

SSL (Secure Sockets Layer) is a technology for securing socket based distributed systems originally developed by Netscape. This is supported by many libraries and languages on most platforms, and whilst currently only a defacto standard is being revised into a formal standard (as an RFC) as TLS (Transport Level Security)

Your assignment this term is to design and implement a small, SSL enabled client server application. It is suggested that a minimal http server (based for example on linuxsocket.org's http-ls server) is developed so that interaction with a standard web browser can be demonstrated.

Main features required:

- Fully chained (i.e you start by creating a certification authority) X.509 certificates for both client & server.
- Client and Server authentication.
- Cipher negotiation.
- Screen based feedback for the steps in and outcome of the SSL handshake.

Additional marks may be gained by demonstrating interaction between your server and a standard web browser in which you have installed a client certificate.

The examples and documentation that are provided on the website for this module use Linux, openssl, and C. To compile the examples from the SSL tutorial you will need to modify the makefile.

Deliverables:

1. Source code for your project, signed off by your tutor to prove (2). 30
2. A demonstration of your project. 20
3. Demonstration against a standard web browser 10
4. A report (4000 words maximum) describing:
 - a. The issues and problems you have encountered in implementing your application. In particular you should explain how you created your certificate chain 10
 - b. A detailed explanation of how SSL/TLS technology works. This should pay particular attention to ciphers, key negotiation and the use of certificates. 30

References:

Thomas, S A; *“SSL & TLS Essentials: Securing the Web”*, Wiley 2000
Rescorla, E; *“An Introduction to openssl programming”*, RTFM Inc. (available from module web page)