Analysing Data-Sensitive and Time-Sensitive Web Applications

Mohammed Alzahrani
mya9@hw.ac.uk

Dependable Systems Group
Heriot Watt University

Structure

- Aims and objectives.
- Modelling of web applications : navigation, authentication and session management.
- Model checking with Spin and UPPAAL.
- Further Work.

Aims

Aims

- To study the usability of formal methods; in particular Model checking for modelling and verifying web applications behaviour.

HERIOT WATT UNIVERSITY

Aims

- To study the usability of formal methods; in particular Model checking for modelling and verifying web applications behaviour.
- We have focused **security** and **navigation** properties along with modelling time constrains.

Objectives

Objectives

- To design a formal model of data-Sensitive and time-sensitive web applications using SPIN.

Objectives

- To design a formal model of data-Sensitive and time-sensitive web applications using SPIN.
- To include time properties in our models to represent realistic web applications.

Objectives

- To design a formal model of data-Sensitive and time-sensitive web applications using SPIN.
- To include time properties in our models to represent realistic web applications.
- To explore the capabilities of different Model Checking Tools.

Web Applications

- Increase in their popularity and usage
- Web applications are used extensively in many areas:

Commerce: online banking, online shopping,
Entertainment: online music,videos,
Interaction: social networks

Web Applications Design

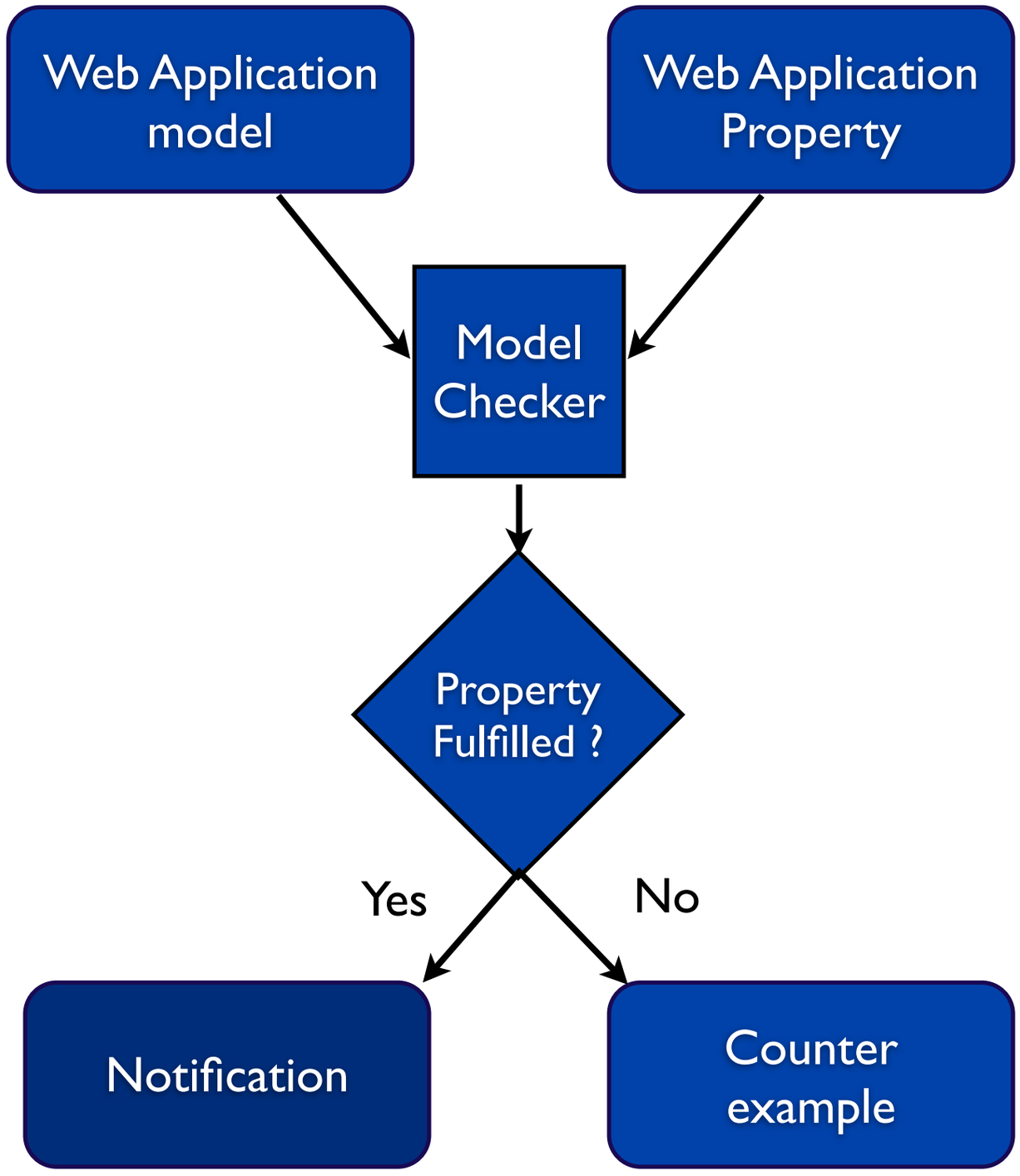
Navigation errors mishandle unexpected user requests.

Global accessibility makes them a target for many malicious users.

HERIOT WATT UNIVERSITY

Model Checking

an automated technique that,given a finite-state model of a system and a logical property, systematically checks whether this property holds for a given initial state in that model.



Edmund M. Clarke. The birth of model checking. In 25 Years of Model Checking, volume 5000 of Lecture Notes in Computer Science, pages 1-26, 2008.

SPIN Model checker

- It is a model checker for the temporal logic LTL.
- Aimed at verification of protocols and software.
- Provides a graphical user interface (ispin) to the model checker and to an interactive simulator.

G.J. Holzmann. The SPIN model checker: Primer and reference manual.
Addison-Wesley Professional, 2004.

Modelling Time

Modelling time is critical to design realistic models of web applications.

Scenarios like (timeouts) and to timestamp messages between communicating parties to avoid attacks.

DTSpin

- We extend our Promela model with discrete time macros.
- This will give us the ability to construct realistic web applications models.

D. Bosnacki, D. Dams. Integrating real time into spin: A prototype implementation.
In FORTE XI / PSTV XVIII, pages 423{438. Kluwer, B.V., 1998.

DTSpin macros

1. **Timers** process is a daemon process that uses ticks to decrease the timer values.

2. **set**(tmr,1); A; B; **expire**(tmr); C

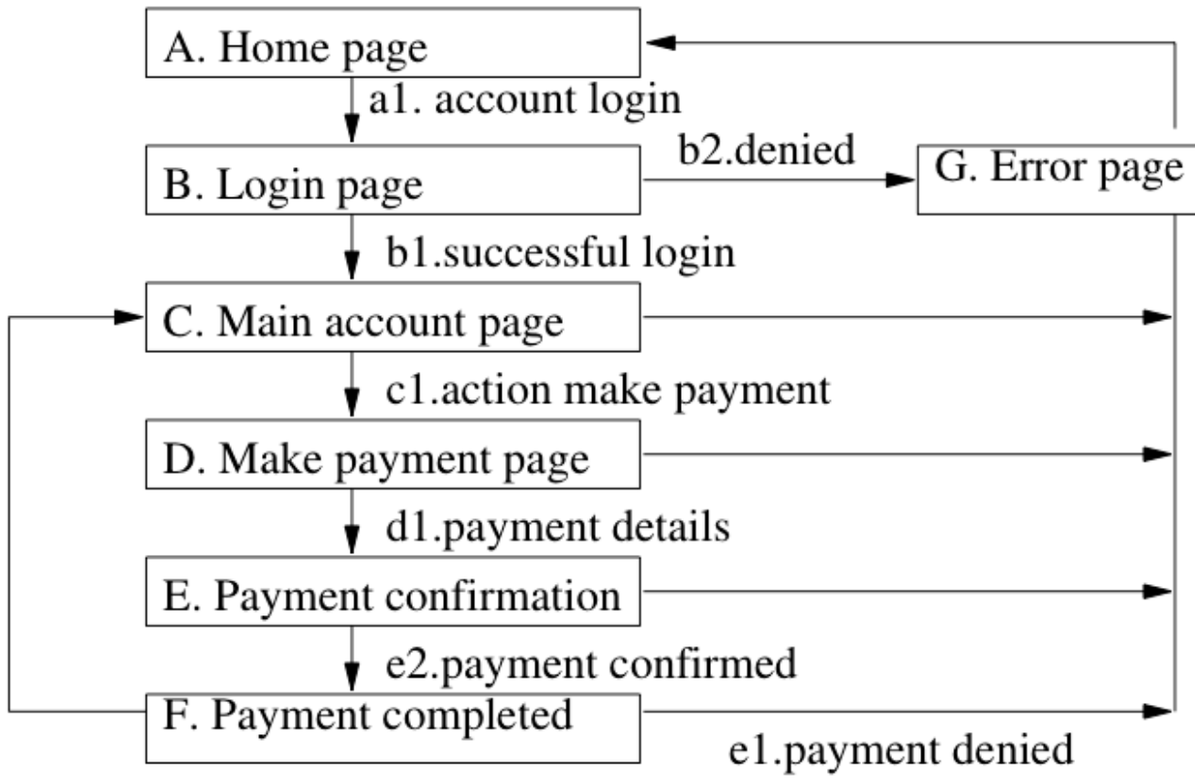
```
#define timer int

#define set(tmr,val) (tmr=val)

#define expire(tmr) (tmr==0) /*timeout*/

#define tick(tmr) if :: tmr>=0 ->
tmr=tmr-1 :: else fi

#define delay(tmr,val) set(tmr,val); expire
(tmr)

#define udelay(tmr) do :: delay(tmr,1) ::
break od


proctype Timers()
{ do :: timeout -> atomic{ tick(tmr1); tick
(tmr2) } od }
```

Modelling Web Applications as Transition System

Web application are modelled using two finite-state automata:

Page Automaton

Internal State Automaton



Investigated how to model:

1.Authentication

2.Session management

3.web navigation

HERIOT WATT UNIVERSITY

Web Pages Automaton

- In page transitions; web pages can be treated as states and; page transitions as a state transition.
- The page transitions is modelled as a finite state-automaton.

Internal-state Automaton

- The Internal states represent the business logic, determined by input values.
- The internal state occurs synchronously with the page transition.

Modelling web navigation

- The home page is reachable from all pages. A user can logo at any stage of the transition
- A page reachable from the home page or account page always has a next page in the transition.
- A user can not reach his account page without going through login page by providing correct credentials first.

Modelling Authentication

- We modelled a security protocol at the start of the session.
- The user input login credentials.

Modelling Session management

- Non-deterministic (Timeouts) are given during the session.

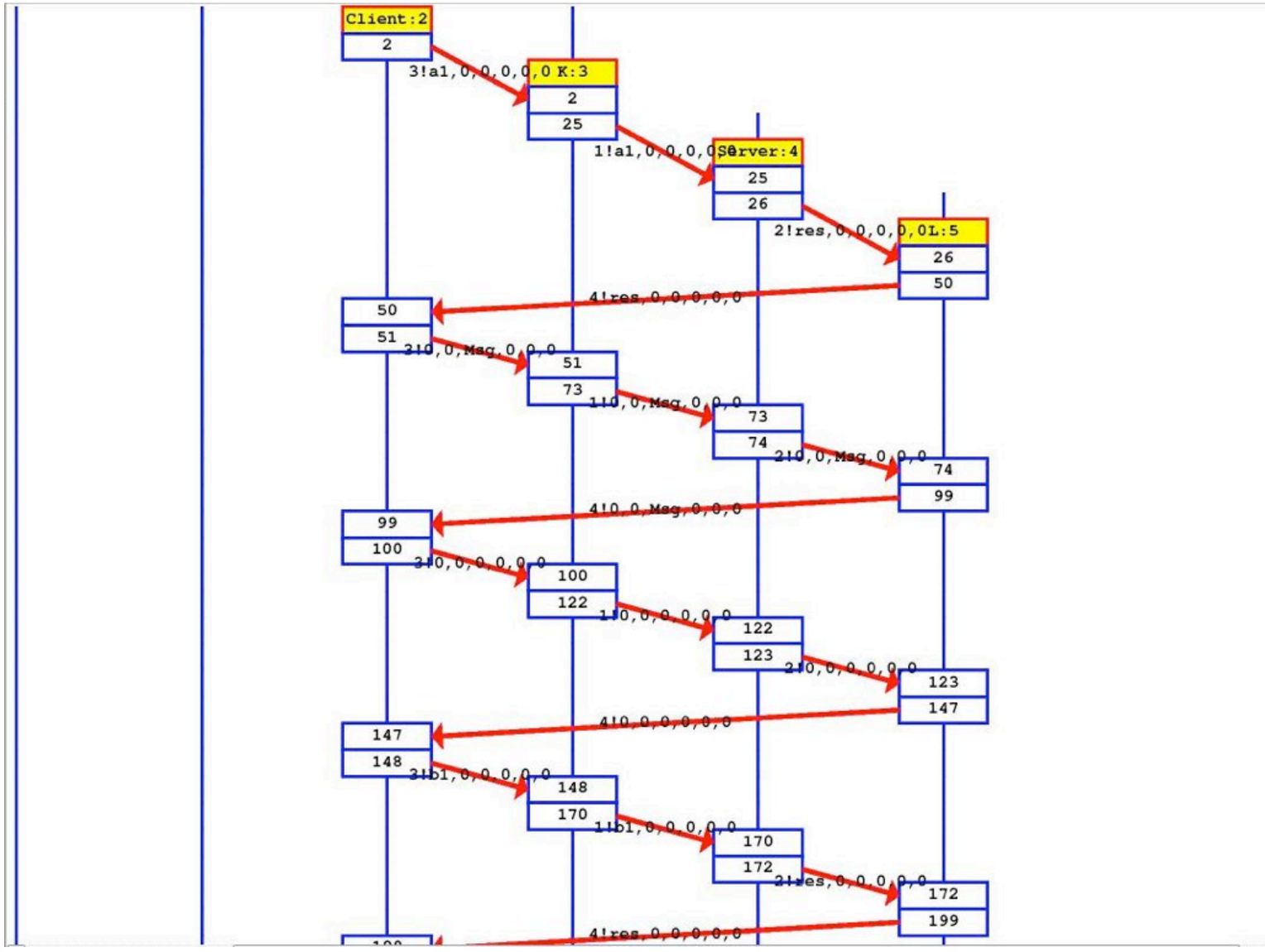
Representation In Promeal

A client is sending a request to login; the server receives it and reply.

```
chan Client ToServer = [0] of {mtype};
chan ServerToClient = [0] of {mtype};
mtype = {loginReq ,ACK};

active proctype Pages()  {

HomePage:
do ::
        if
        :: Client ToServer ! loginReq ->
ServerToCilent ? ACK->goto loginPage;
         fi;
od; }

active proctype InternalState() {
do ::
       if
       :: ClientToServer ? loginReq ->
atomic {ServerToChanel!ACK;goto
SloginPage};
fi;
od;
}
```

Model Simulation

An optimal Client-Server communication, the simulation chart gives an instance see of the model.



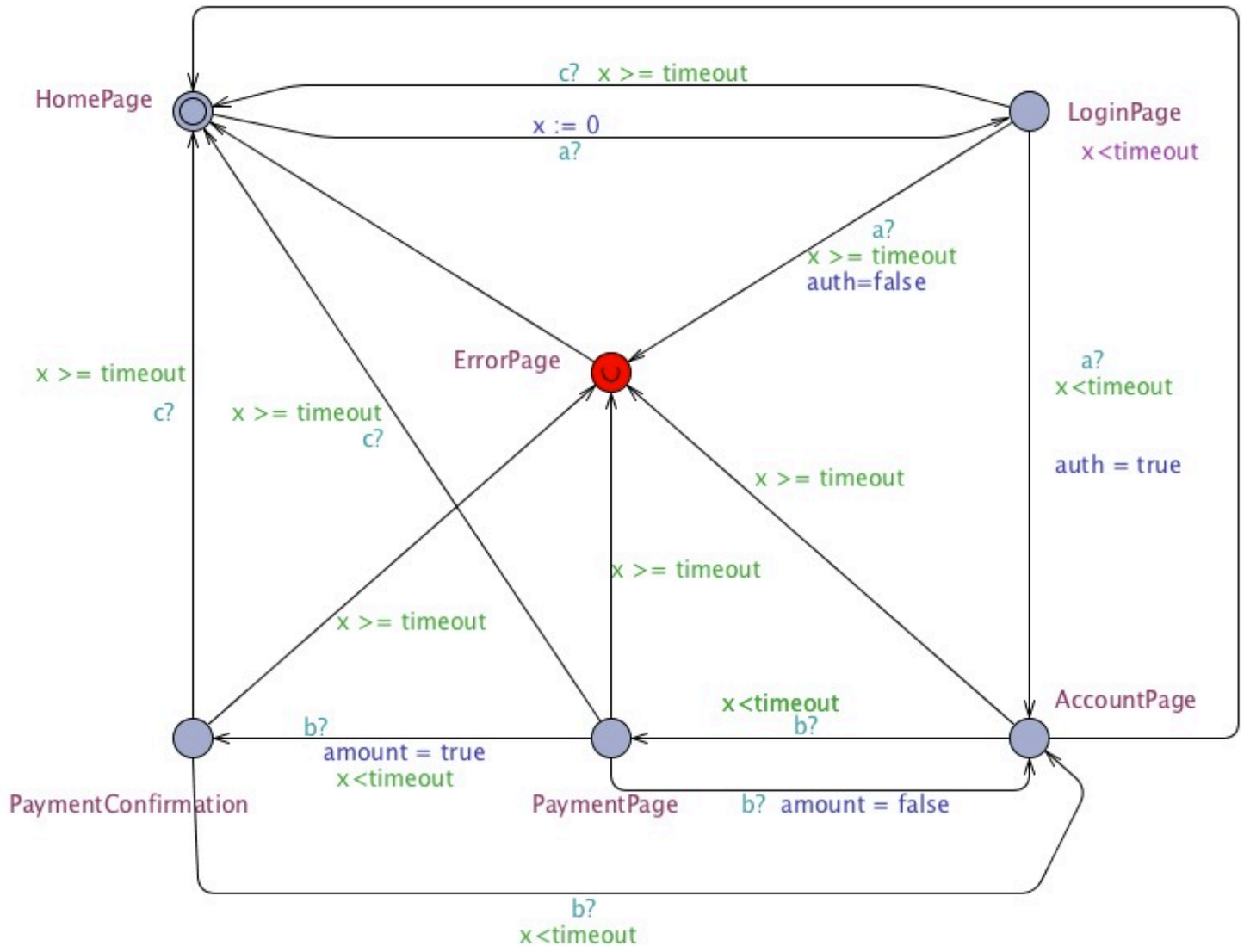
UPPAAL model checker

- Appropriate for systems that can be modeled as a collection of non-deterministic processes with finite control structure and real-valued clocks (i.e. timed automata
- Models are defined graphically.

Gerd Behrmann and Re David and Kim G. Larsen: A tutorial on uppaal.
In Springer, pages 200-236.2004.

Modelling with UPPAAL

- Initial Web pages process.



SPIN vs UPPAAL

- Design time in UPPAAL is less than building the Promela model.
- Same model processes are defined in both tools.
- In Spin early modelling faults can be detected via the “ Message Sequence Charts”

Further Work

- Compare the differences between compromised model and a secure model by analysing The sequence of actions and time stamps.
- We will use UPPAAL to compare and validate our results.

HERIOT WATT UNIVERSITY

Thanks !

HERIOT WATT UNIVERSITY

Questions