Formal Verification for Legal Privacy Requirements

Michael May, Carl Gunter, Insup Lee

September 2005

Premise

- Two universes: legal privacy rules and access control
- Formal tools exist for access control
- How far can we get using access control to model legal privacy rules?
  - Discover problems?
  - Do verification?
  - Compare legal texts?
- Can we use formal tools to analyze and model legal requirements?

Related Work

- Access control theory
  - Graham/Denning [Graham and Denning 1972]
  - HRU [Harrison, Ruzzo, Ullman 1976]
  - Originator control (ORCON) [Graubart 1989]
  - Privacy Systems [Gunter, May, and Stubblebine 2004]
- Privacy of personal information regulations
  - Health Information Portability and Accessibility Act's Standards for Privacy of Individually Identifiable Health Information (USA)
  - 2002/58/EC Concerning the processing of personal data and the protection of privacy in the electronic communications sector (EU)
- Computer science and legal regulations
  - Security Policy for Clinical Information Systems [Anderson 1996]
  - Cassandra [Becker and Sewell 2004]

Outline

- Background
  - Access Control
    - HRU
  - HIPAA
- Translation
  - Examples
- Verification
  - Model design and implementation
  - Examples and results

Access control

- Classical access control systems use access control matrices
  - Graham/Denning, HRU
  - Subjects, objects, permissions
- Policy rules define legal operations

	Alice	Bob	F1	F2	F3	F4
Alice		control	r		w	own
Bob			r,x	r		w

Policy Example (HRU)

Create (process, file)

    create object file

    insert own **into** (process, file)

end

Confer (owner, friend, file)

    if own **in** (owner, file)

    then insert r **into** (friend, file)

end

HRU

- HRU structure for access control matrix policy
- *Operations* are primitives in the matrix
  - Enter right, delete right, create object, etc.
- *Commands* are sets of operations with an optional guard
  - Guards can refer only to information available to the system
  - Commands run transactionally
- Result: General problem of rights leakage is undecidable [HRU 1976]
  - Decidable only if commands are limited to one operation
  - Model checking and state exploration can help

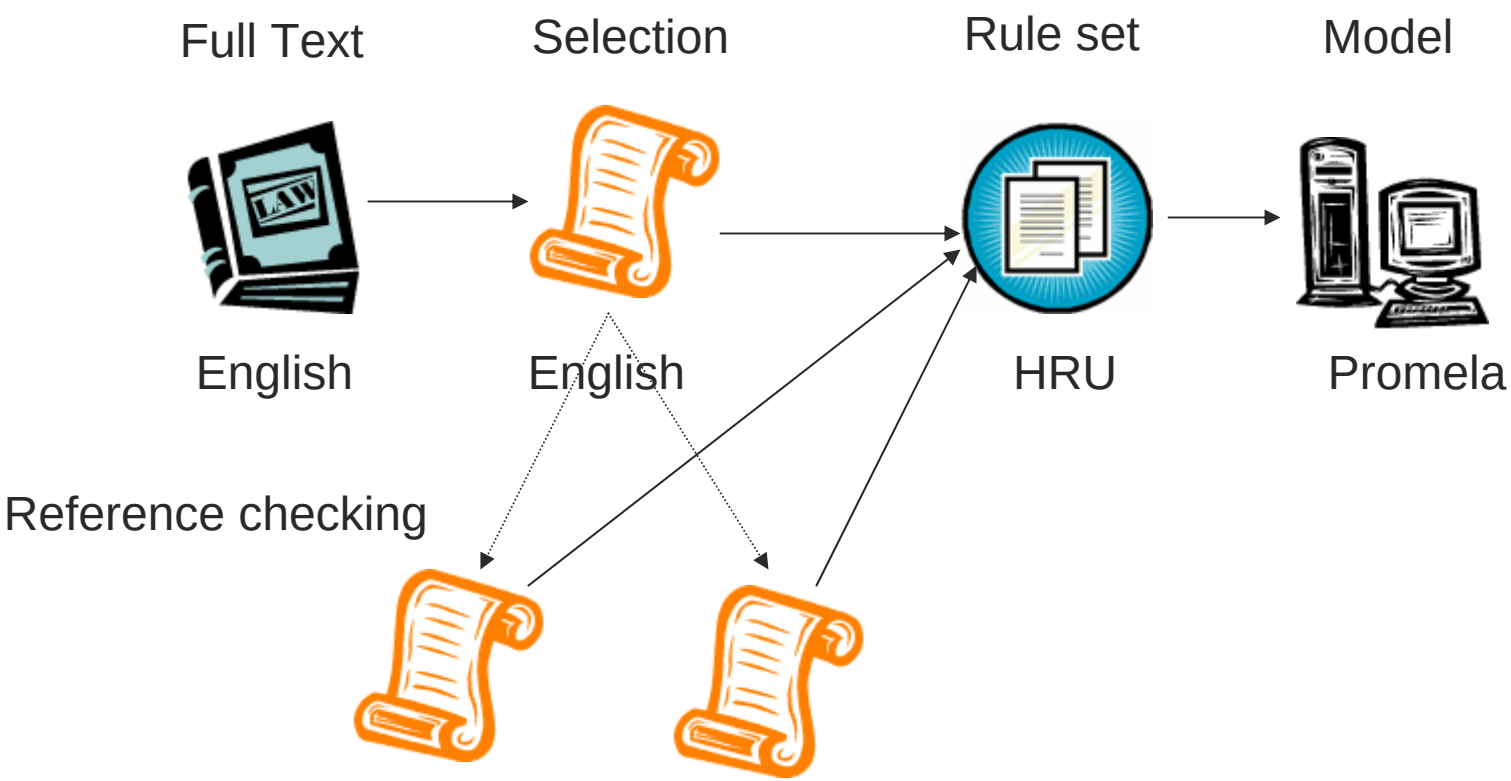
HIPAA

- 1996 US regulation on management of private health information
- Several comment and rewrite cycles with significant revisions and restructuring
  - Privacy and medical experts critiqued the law at each step
- Final rule has a 25 page section of privacy rules
  - Medical facilities must have a Privacy Officer whose job is creating and enforcing HIPAA policies

Outline

- Background
  - Access Control
    - HRU
  - HIPAA
- Translation
  - Examples
- Verification
  - Model design and implementation
  - Examples and results

Translation steps



Example 1: Easy clause

[2003] 164.506(c)(1): A covered entity may use or disclose protected health information for its own treatment, payment, or health care operations.

- What do we need to convert this to an HRU style command?
  - Actors who acts on behalf of a covered entity
  - Objects with ownership labels and subject labels
  - Method of indicating the purpose of a command
  - Method of managing permissions on objects after they are disclosed

Example 1: Translation

164.506(c)(1): A covered entity may use or disclose protected health information for its own treatment, payment, or health care operations.

- Let a be an agent of a covered entity (hospital, doctor's office, etc)
- Let f be a file which contains protected health information

HRU Rules:

```
treatment506c1 (a, subject,
recipient, f, evidence )
If own in (a, f)
And recipient == a
Then insert treat in (a, f)
End
treatmentDisclose506c1 (a,
subject, recipient, f, evidence)
If own in (a, f)
Then insert treat-disclose in
(a, f)
And copy f as f'
And insert own in (recipient,
f')
End
```

Example 1: Discussion

- Each command is aware of the actor (attempting to) perform it
- Disclosure is performed with a copy command that preserves object properties
- We need a way to load in roles into the access control matrix
- We put the purpose of a command in its title
  - Purpose is essential to decision making, but is not something the system can determine without a testimonial

Example 2: Complex clause

[2003] 164.506(a): Standard: Permitted uses and disclosures. Except with respect to uses or disclosures that require an authorization under 164.508(a)(2) and (3), a covered entity may use or disclose protected health information for treatment, payment, or health care operations as set forth in paragraph (c) of this section, provided that such use or disclosure is consistent with other applicable requirements of this subpart.

- What more do we need to convert this to an HRU style command?
  - A notion of pointing to another clause
  - A way to determine what obligations are required by another section
  - A way to resolve vague references (“consistent with other applicable requirements of this subpart”)

Example 2: Translation

164.506(a): Standard: Permitted uses and disclosures. Except with respect to uses or disclosures that require an authorization under 164.508(a)(2) and (3), a covered entity may use or disclose protected health information for treatment, payment, or health care operations as set forth in paragraph (c) of this section, provided that such use or disclosure is consistent with other applicable requirements of this subpart.

- Let a be an agent of a covered entity (hospital, doctor's office, etc)
- Let f be a file which contains protected health information

HRU Rule:

```
treatment506a (a, subject, recipient, f, evidence)
If !AsIn508a2 (a, subject, recipient, f, evidence)
And !AsIn508a3 (a, subject, recipient, f, evidence)
And "consistent with other applicable requirements of this subpart" in evidence
And AllowedAsIn506c (a, subject, recipient, f, evidence)
Then insert treat in (a, f)
end
```

Example 2: Translation

- **IF** NeedAuthAsIn508a2 (actor, subject, recipient, file, evidence)
  if AsIn508a2i (actor, subject, recipient, file, evidence)
  or AsIn508a2ii (actor, subject, recipient, file, evidence)
  then return false
  else return true
  end
- **IF** NeedAuthAsIn508a3 (actor, subject, recipient, file, evidence)
  if AsIn508a3i (actor, subject, recipient, file, evidence)
  or AuthValidAsIn508a3ii (actor, subject, recipient, file, evidence)
  then return false
  else return true
  end

- **IF** AllowedAsIn506c (actor, subject, recipient, file, evidence)
  if AllowedAsIn506c1 (actor, subject, recipient, file, evidence)
  or AllowedAsIn506c2 (actor, subject, recipient, file, evidence)
  or AllowedAsIn506c3 (actor, subject, recipient, file, evidence)
  or AllowedAsIn506c4 (actor, subject, recipient, file, evidence)
  or AllowedAsIn506c5 (actor, subject, recipient, file, evidence)
  then return true
  else return false
  end

Example 3: Testimonials

[2000] 164.506(a)(3)(i) A covered health care provider may, without prior consent, use or disclose protected health information created or received under paragraph (a)(3)(i)(A)-(C) of this section to carry out treatment, payment, or health care operations: …

(C) If a covered health care provider attempts to obtain such consent from the individual but **is unable to obtain such consent due to substantial barriers to communicating with the individual, and the covered health care provider determines, in the exercise of professional judgment, that the individual's consent to receive treatment is clearly inferred from the circumstances.**

- The system can not determine whether this condition is true without a testimonial from someone

Example 3: Translation

164.506(a)(3)(i) A covered health care provider may, without prior consent, use or disclose protected health information created or received under paragraph (a)(3)(i)(A)-(C) of this section to carry out treatment, payment, or health care operations:

…

(C) If a covered health care provider attempts to obtain such consent from the individual but **is unable to obtain such consent due to substantial barriers to communicating with the individual, and the covered health care provider determines, in the exercise of professional judgment, that the individual's consent to receive treatment is clearly inferred from the circumstances.**

HRU Rule:

```
IF AsIn506a3iC (a, subject,
recipient, f, evidence )
If attempted in (a, f)
And consent not in (subject, f)
And “barriers to communication”
in evidence
And “professional judgment” in
evidence
Then return true
Else return false
End
```

- Let a be an agent of a covered entity (hospital, doctor’s office, etc)
- Let f be a file which contains protected health information

Discussion

- Two major issues in translation
- Creating AsIn rules
  - Pointers that reference the guard of a paragraph
  - Need to separately name the guard of a paragraph from its body
  - Resolving the meaning of vague statements
- Conditions that can not be checked by inspecting system state or the access matrix
  - We need a testimonial from a human
  - Testimonials come from actors in the system and must be logged to be auditable

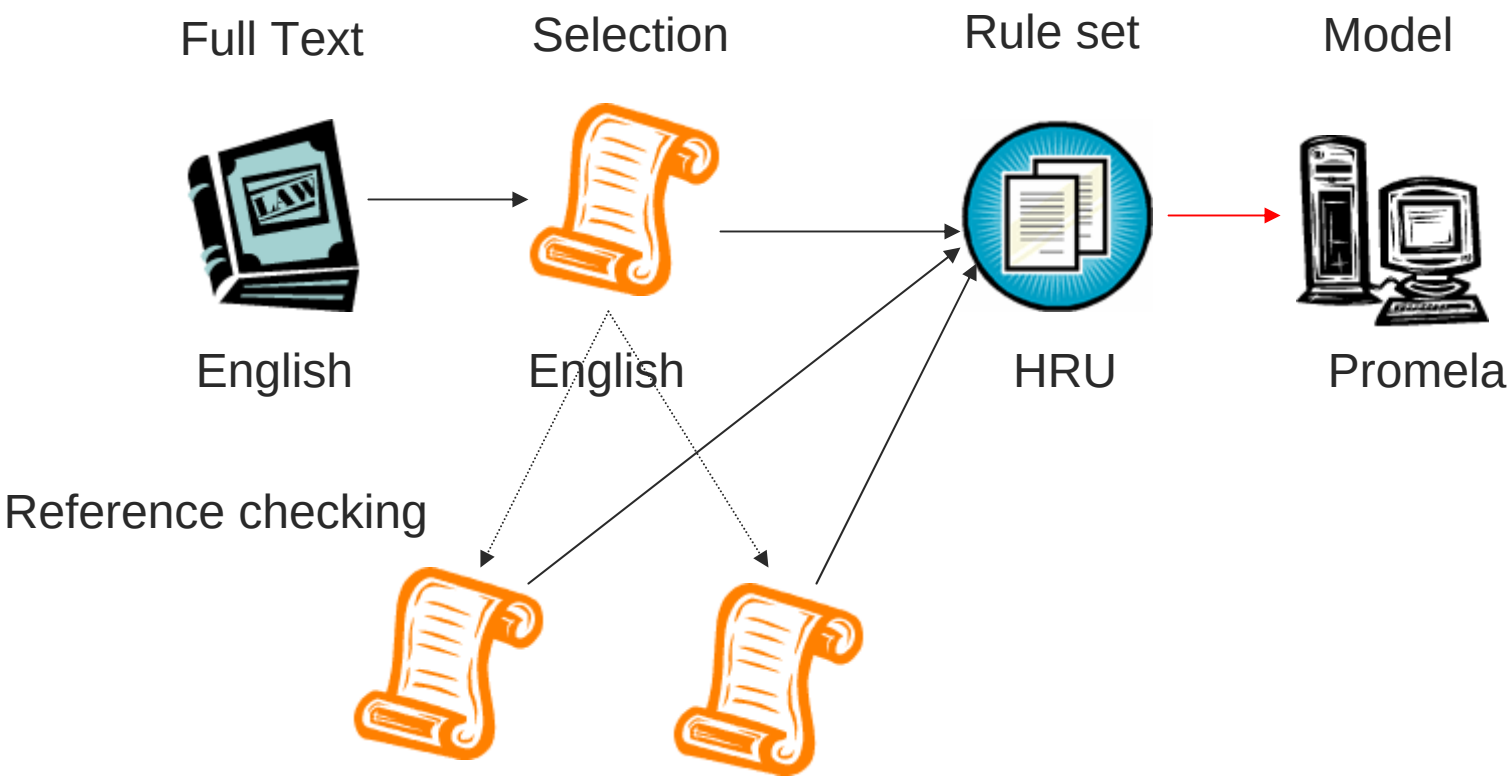
Creating the rule sets

- Using above techniques we translated one section (164.506) on consent for disclosure
  - 2000 and 2003 versions of the rules very different
  - Chasing references lead to including a large section of text
- Rules designed to follow the structure of the law closely
  - Semi-automation of the process in the future
- Rule set size
  - 2000: 50 + 42 (by ref) = 92 rules
  - 2003: 26 + 32 (by ref) = 58 rules

Outline

- Background
  - Access Control
    - HRU
  - HIPAA
- Translation
  - Examples
- Verification
  - Model design and implementation
  - Examples and results

Translation steps



Model

- Modeled the rule set in Spin
  - Trace the path that lead to specific valid and invalid states
  - Count the steps needed to reach states
  - Valid and invalid states are designated by experts in health care and privacy activists
    - Balance between stakeholders invariants is essential in creating a fair regulation

Model Design

- Rules become processes
- Channels to communicate between processes
- Access matrix is a 2-d array
  - Entries in the array are record sets with bit flags for each possible permission
  - Principals are names with corresponding integer representation for indexing
- Roles managed by having a permission “member” on group objects

Model Example 1

```
treatmentDisclose506c1 (a, subject, recipient, f, evidence)
If AllowedAsIn506c1(a, s, r, f, e)
Then insert treat-disclose in (a, f)
And copy f as f'
And insert own in (recipient, f')
End
```

```
active proctype treatmentDisclose506c1 (/* a,  s,  r,  f,  f_new*/){
bool result = false;
bool temp;

do
:: treatmentDisclose506c1_chan?request(_) ->
   AllowedAsIn506c1_chan!request(true);
   AllowedAsIn506c1_chan?response(temp);
   result = temp;

   if
   :: result ->
      /* the new object is the top item; make sure we do not run out of room */
      atomic{assert(top < N); f_new = top; top = top + 1;}

      m.mat[a].obj[f].treatDisclose = 1; m.mat[s].obj[f_new].subject = 1;
      m.mat[a].obj[f_new].originator = 1; m.mat[r].obj[f_new].own = 1;
   :: else -> skip;
   fi;
   treatmentDisclose506c1_chan!response(result);
od}
```

Model Example 1

```
active proctype AllowedAsIn506c1 (/*a,  s,  r,  f*/){
bool result = false;
bool temp;

do
:: AllowedAsIn506c1_chan?request(_) ->
    temp = (m.mat[a].obj[f].own == 1);
    result = temp && (evidence.own_use == 1);

    AllowedAsIn506c1_chan!response(result);
od}
```

Model Example 2

```
treatment506a (a, subject, recipient, f, evidence)
If !AsIn508a2 (a, subject, recipient, f, evidence)
And !AsIn508a3 (a, subject, recipient, f, evidence)
And “consistent with other applicable requirements of this subpart” in evidence
And AllowedAsIn506c (a, subject, recipient, f, evidence)
Then insert treat in (a, f)
End
active proctype treatment506a(/*a, s, r, f*/){
bool result = false;
bool temp;
do
:: treatment506a_chan?request(_) ->
   NeedAuthAsIn508a2_chan!request(true);
   NeedAuthAsIn508a2_chan?response(temp);
   result = !temp;
   NeedAuthAsIn508a3_chan!request(true);
   NeedAuthAsIn508a3_chan?response(temp);
   result = result && !temp;
   temp = (evidence.consistent_with_other_applicable_requirements_of_this_subpart==1);
   result = result && temp;
   AllowedAsIn506c_chan!request(true);
   AllowedAsIn506c_chan?response(temp);
   result = result && temp;
   m.mat[a].obj[f].treat = (result -> 1:m.mat[a].obj[f].treat);
   treatment506a_chan!response(result);
od
```

Model Validation

- Model Validation
  - Simple sanity checks to ensure reachability of single step moves
  - More complex multi-step verification runs
- We can use the model to detect problems in the law
  - Comments on the 2000 version consent rules lead to a complete rework in the 2003 version

Verification using the rule sets

- We use Spin to find the problems previously detected by manual inspection
  - Ex: Ambulance workers must obtain consent for services they did for unconscious patients after the fact
  - Ex: Hospitals which usually do pre-operation preparations before procedures can not do so without the patient coming to sign a special designator
  - Ex: Doctors who render remote diagnoses can not do so without having a special paper consent form sent or faxed to them first.
- Theorem Example
  - Given an object f about principal Paula (patient). Principal Dan (doctor) can not gain any access permissions on f without getting consent from Paula first (or after the fact in case of emergency).

Theorem example code

```
m.mat[Dan].obj[health_care_provider_group].member = 1;
m.mat[Dan].obj[covered_entity_group].member = 1;
m.mat[Paula].obj[f].subject = 1;
m.mat[Paula].obj[f].consent = 0;

#define inv m.mat[Dan].obj[f].treat ==0
/** Formula As Typed: []inv
* The Never Claim Below Corresponds
* To The Negated Formula !([]inv)
* (formalizing violations of the original)*/
never {
/* !([]inv) */
T0_init:
    if  :: (! ((inv))) -> goto accept_all
        :: (1) -> goto T0_init
    fi;
accept_all: skip}
```

Research goals

- Future goal: Compare the rule sets and discover differences
  - Evaluate if anything else changed between the versions
- Verification of systems that implement the rules
  - Gain a rigorous definition of legal compliance
  - Aid correct implement through verification and testing
  - Motivate design of laws that are technical specifications to be verifiable technologically
- Processing from natural language to rule sets

Conclusion

- Using computer access control techniques to understand legal regulations
  - Translating one to the other reveals similarities between them
  - Differences require us to rethink some theories of computer access control
- Success in modeling the sections of the regulation closest to access control rules
  - Some sections are not addressable
    - Ex: Typographical rules for writing a privacy practices declarations
- Research goal is to use the models to better understand the implementation and evolution of regulations