1 Apr 2003, Baltimore

Introducing Cyberlogic[a]

Harald Rueß and Natarajan Shankar
email: ruess,shankar@csl.sri.com
url: http://www.csl.sri.com/~ruess

Computer Science Laboratory
SRI International
Menlo Park, CA

[a]Supported by NSF and SRI International. Opinions expressed are those of the authors.

What is Cyberlogic?

Cyberlogic is a logic framework for **evidential transactions** built on the public-key infrastructure.

Key ideas.

1. Public keys correspond to authorizations.

2. Digital certificates are proofs (evidence).

3. Protocols are distributed logic programs.

Simple foundation for a number of applications such as digital government, access control, digital rights management.

This talk is an informal overview of some preliminary ideas on Cyberlogic.

Who Needs Proof?

You get pulled up on a highway while driving to work. The patrolman checks that you have a valid drivers' license and registration. He tells you that you have a broken tail-light and gives you a ticket that will be cancelled if you have it fixed by an authorized mechanic within 10 days. You have the tail-light fixed that weekend. The mechanic issues a digital certificate that the light has been fixed. You mail this to the CHP who respond with a digital certificate indicating that the ticket has been cancelled.

The evidence here, consisting of the drivers' license, registration, the completion of repair, and the ticket cancellation, can be represented and processed in electronic form.

Attestations

Apart from the usual assertions, like $(\forall x : \exists y : x < y)$, Cyberlogic contains *attestations*

$$K \rhd A$$

This means that public key K attests formula A.

Thus, public keys correspond to *authorizations* (and not to identities)

Examples.

$$K_{DMV} \rhd hasDriversLicense(\mathit{name}, \mathit{SSN}, \mathit{DOB}, \mathit{expiry})$$
$$K_{INS} \rhd hasUSvisa(\mathit{name}, \mathit{passport\_number}, \mathit{type}, \mathit{validity})$$
$$K_{WWBank} \rhd hasAccount(\mathit{name}, \mathit{account\_number}, \mathit{balance}, \mathit{time})$$
$$K_{INS} \rhd isResident(\mathit{name}, \mathit{SSN}, \mathit{expiry})$$

Certificates as proofs

If $\{c\}_K = A$, then c proves $K \rhd A$.

$$c \ : \ K \rhd A$$

Does not necessarily mean that A is valid.

Cyberlogic = intuitionistic logic + attestations

- $\langle p, q \rangle \ : \ A \wedge B$ if $p : A$, $q : B$
- $inl(p) \ : \ P \vee Q$ if $p : P$, and $inr(q) \ : \ P \vee Q$ if $q : Q$
- $p \ : \ A \Rightarrow B$, if for all $q : A$, $p(q) \ : \ B$
- $p \ : \ \forall x.\ A$, if for all a, $p(a) \ : \ A[a/x]$
- $(a, p) \ : \ \exists x.\ A$ if $p \ : \ A[a/x]$

Attestations

Some reduction rules for $K \rhd A$:

$$
\begin{aligned}
K \rhd (A \wedge B) &\equiv (K \rhd A) \wedge (K \rhd B) \\
K \rhd (A \vee B) &\equiv (K \rhd A) \vee (K \rhd B) \\
K \rhd (A \Rightarrow B) &\equiv A \Rightarrow (K \rhd B) \\
K \rhd (\forall x : A) &\equiv (\forall x : K \rhd A) \\
K \rhd (K \rhd A) &\equiv K \rhd A
\end{aligned}
$$

Direct vs. Indirect Evidence

A *direct* proof for an attestation is given by a digital certificate.

If $\{c\}_K = A$, then c directly proves $K \rhd A$.

This stronger claim is written as $K :\!\rhd A$.

$(K :\!\rhd A) \Rightarrow (K \rhd A)$.

On the other hand, if the proof of $K \rhd B$ follows from those of $K :\!\rhd (A \Rightarrow B)$ and A, then we only have the indirect claim $K \rhd B$.

This has implications when an authorization is *delegated*.

Authorization and Delegation

$K \rhd A$ does not imply the validity of A.

Such validity only follows when K possesses authority over A.

The authority of a key must itself be attested by a certification authority.

Single certification authority K_{CA}, anything attested by this certification authority is valid. That is,

$$(K_{CA} \rhd A) \Rightarrow A.$$

The certification authority can selectively authorize another K through

$$K_{CA} \rhd (K \rhd p(\overline{S})) \Rightarrow p(\overline{S}).$$

Authorization and Delegation (Cont.)

If the intention is that this authority over $p(\overline{S})$ is not to be further delegated, then the certification authority must restrict the attestation to be direct as in

$$K_{CA} \rhd (K :\!\rhd p(\overline{S})) \Rightarrow p(\overline{S}).$$

Unbounded delegation of an authority from K to K' is carried out by means of the attestation

$$K :\!\rhd (K' \rhd A) \Rightarrow A.$$

Bounded delegation is performed by the attestations

$$\begin{aligned} \mathcal{D}_1(K, K', A) &:= K :\!\rhd ((K' :\!\rhd A) \Rightarrow A) \\ \mathcal{D}_2(K, K', A) &:= K :\!\rhd (\forall K''.\ (K'' :\!\rhd A) \wedge K' \rhd (K'' :\!\rhd A) \Rightarrow A) \Rightarrow A \end{aligned}$$

Bounded delegation can be iterated to an arbitrary finite depth.

Executing Cyberlogic

Protocols are implemented as distributed logic programs.

Logic programs are hereditarily Harrop Cyberlogic formulas.

Programs

$$D \quad ::= \quad Atom \mid G \Rightarrow Atom \mid \forall x.\ G \mid D \wedge D$$

Queries

$$G \quad ::= \quad Atom \mid G \wedge G \mid G \vee G \mid D \Rightarrow G \mid \exists x.\ G \mid \forall x.\ G$$

Anwering queries. In each execution step, either a

- Goal is decomposed into subgoals, or
- Goal is sent from a supplicant to appropriate authority, or
- Reply to a goal in the form of a proof is sent back to the supplicant.

Scenario: Accessing Medical Records

Policy of a.

$$
\begin{aligned}
a_1 &: K_a \rhd isHospital(b) \\
a_2 &: K_a \rhd isHospital(c) \\
a_3 &: \forall X, Y.\ K_a \rhd isPhysician(X, Y) \Rightarrow K_a \rhd readMedRec(X, Y) \\
a_4 &: \forall X, Y, Z.\ K_a \rhd isHospital(Z) \Rightarrow \mathcal{D}_1(K_a,\ Z,\ isPhysicianOf(X, Y)) \\
a_5 &: \forall H, Z_1, Z_2.\ Z_1 \neq Z_2 \wedge Z_1 \neq a \wedge Z_2 \neq K_a \wedge \\
&\quad K_a \rhd (isHospital(Z_1) \wedge isHospital(Z_2)) \Rightarrow \\
&\quad \mathcal{D}_1(K_a,\ Z_1, Z_2,\ isHospital(H))
\end{aligned}
$$

Policy of b.

$$
\begin{aligned}
b_1 &: K_b \rhd isHospital(a) \\
b_2 &: K_b \rhd isHospital(b) \\
b_3 &: K_b \rhd isPhysicianOf(alice, peter)
\end{aligned}
$$

Policy of c.

$$c_1 : K_c \rhd isHospital(b)$$

Scenario: Accessing Medical Records (Cont.)

$$?_0 \quad : \quad K_a \rhd readMedRec(alice, peter)$$

$?_0 \equiv a_3(alice)(peter)(?_1)$

$$?_1 \quad : \quad K_a \rhd isPhysician(alice, peter)$$

$?_1 \equiv a_4(alice)(peter)(Z)(?_2)(?_3)$

$$?_2 \quad : \quad K_a \rhd isHospital(Z)$$
$$?_3 \quad : \quad Z \rhd isPhysician(alice, peter)$$

$?_3 \equiv b_3$, $?_2 \equiv a_5(b)(Z_1)(Z_2)(?_5)(?_6)(?_7)(?_8)(?_9)(?_{10})$

$$\langle ?_5, ?_6, ?_8 \rangle \quad : \quad Z_1 \neq Z_2 \wedge Z_1 \neq a \wedge Z_2 \neq a$$
$$?_9 \quad : \quad K_a \rhd (isHospital(Z_1) \wedge isHospital(Z_2))$$
$$?_{10} \quad : \quad Z_1, Z_2 \rhd isHospital(b)$$

$?_9 \equiv \langle a_1, a_2 \rangle$, $?_{10} \equiv \langle b_2, c_1 \rangle$, and $?_5$, $?_6$, $?_8$ from background theory.

Distributed Proof Search!

Scenario: Accessing Medical Records (Cont.)

Certification authority.

$$ca_1 \quad : \quad CA \ :\rhd K_{hmo} \ \rhd \ isHospital(x) \Rightarrow isHospital(x)$$

Policy of HMO.

$$hmo_1 \quad : \quad T(a) \wedge T(b) \wedge T(c)$$

$$hmo_2 \quad : \quad \forall K, x.\ T(K) \Rightarrow (K \ :\rhd isHospital(x)) \Rightarrow K_{hmo} \ \rhd \ isHospital(x)$$

Authentication based on Needham–Schroeder protocol

Agent a wishes to establish an mutually authenticated session with agent b.

Query.

$$\forall N_a :$$
$$(\forall N_b : K_b \rhd Channel(a, b, N_a, N_b) \Rightarrow K_a \rhd Channel(a, b, N_a, N_b))$$
$$\Rightarrow K_b \rhd init(a, b, N_a)$$

Program.

$$\forall N_a :$$
$$(\forall N_b : K_b \rhd Channel(a, b, N_a, N_b) \Rightarrow K_a \rhd Channel(a, b, N_a, N_b))$$
$$\Rightarrow K_b \rhd init(a, b, N_a)$$

An early Cyberlogic manifesto...

If electronic mail systems are to replace the existing paper mail system for business transactions, "signing" an electronic message must be possible. The recipient of a signed message has proof that the message originated from the sender. This quality is stronger than mere authentication . . . ; the recipient can convince a "judge" that the signer sent the message. To do so, he must convince the judge that he did not forge the signed message himself. In an authentication problem the recipient does not worry about this possibility, since he only wants to satisfy *himself* that the message came from the sender.

(R.L. Rivest, A. Shamir, and L. Adleman, 1977)

Certifying Time

Time often needed to establish that a piece of evidence was produced before or after a certain time.

Trusted time source T that attests that a specific time instant has elapsed.

T broadcasts time ticks of the form

$$c_t : K_T :\triangleright time(t).$$

Certificates c_t can then be embedded in other certificates to establish that a piece of evidence was produced following time t.

Thus, if $\{c\}_K = \langle A, c_t \rangle$ where $c_t : K_T :\triangleright time(t)$, then $c : K :\triangleright_t A$.

Certifying Time (Cont.)

To demonstrate that a claim $K \rhd A$ was produced prior to time t, we once again rely on the trusted time source to timestamp the claim.

The attestation $K_T :\rhd^t A$ indicates that the trusted time source attests to A being verifiable prior to time t.

The certificate c is evidence for $K_T :\rhd^t A$ if $\{c\}_{K_T} = \langle A, t \rangle$.

Since T is a trusted time source, its timed attestations can be taken as valid so that

$$K_T :\rhd^t (K \rhd A) \Rightarrow K \rhd^t A \ .$$

Revocation

Timestamping can be used to ensure that revocable certificates have not been revoked at a given time.

This can be done by ensuring that the authority granted by the certification authority is fresh, as given by

$$T :\triangleright^{t} (CA :\triangleright (K \triangleright A) \Rightarrow A).$$

However, this mechanism does not specify that the authority is revocable.

Revocation (Cont.)

A revocable delegation of authority from K to K' can be specified as

$$K :\triangleright (K' \triangleright A[t'] \wedge K \triangleright NR(K', t) \wedge t' < t \Rightarrow A[t']).$$

This formula asserts that if K attests that K's authority has not been revoked up to time t, then any attestation by K' of the statement $A[t']$ for $t' < t$ is tantamount to an attestation by K of $A[t']$.

Another mechanism for ensuring the freshness of evidence is through the use of *nonces*.

One participant in a protocol could challenge another participant to produce evidence with a newly generated number, the nonce, so as to ensure the recency of the evidence.

Related Work

Trust management systems for authorization in decentralized environments.

SPKI/SDSI. (Ellison, Frantz, Lampson, Rivest, Thomas, Yvonen; 1997)

Many logical formalizations of **SPKI/SDSI**

PolicyMaker/KeyNote. (Blaze,Feigenbaum,Lacy; 1996)

Delegation Logic. (Li, Grosof, and Feigenbaum; 2000)

Proof-Carrying Authorization. (Appel, Felten)

Deontic Logics. (Glasgow, McEwan; 1988)

...

Conclusions

Enabling foundation for building and analyzing protocols that involve the exchange of electronic forms of evidence.

Evidence encoded by means of numbers using digital certificates.

Keys are used to identify specific *authorities*.

Statements such as $P(s)$ are signed by private keys K to obtain a certificate c, so that c is evidence for the claim that K attests that s has property P.

Verified by decrypting c with the corresponding public key $\bar{K}$ to see if it yields $P(s)$.

Protocols are distributed logic programs that gather evidence by using both ordinary predicates and digital certificates.

Conclusions (Cont.)

Time Certification

Gathering of evidence for non-revocation

Simple building blocks for constructing a rich variety of services in a variety of domains

Applications.

- Digital government
- Digital rights management
- Access control in computer systems
- Workflows

Work in Progress . . .

- Design of distributed Cyberlogic interpreter
- Possible-worlds semantics
- Protocols restrict possibility, knowledge
- Correctness of protocols, Termination, Complexity
- Higher-order Cyberlogic