ACON: Activity-Centric Access Control for Social Computing

Aug. 25, 2011
International Conference on Availability, Reliability and Security

Jaehong Park, Ravi Sandhu, Yuan Cheng
Institute for Cyber Security
University of Texas at San Antonio

Social Computing

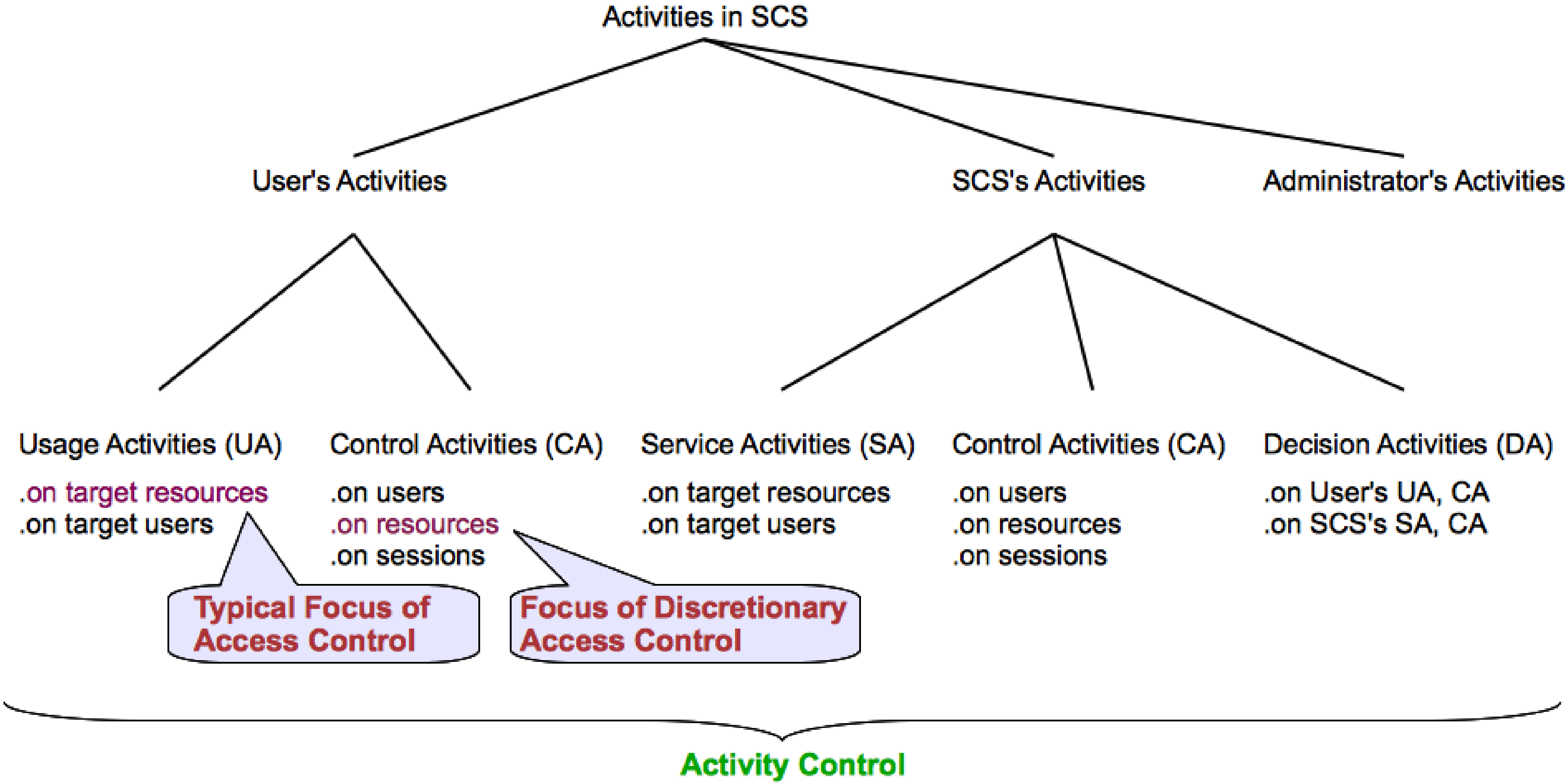
- Characteristics
  - Social computing systems (SCS) provide services to promote information sharing by utilizing user activity information and shared contents
    - Best seller, friends recommendation, friend activity notification, location-based service
  - Both user and SCS provide/access information to be shared
  - A user wants to control other user's or SCS's activities against shared information or users related to her
  - User wants to protect their privacy
  - Both resource and user as a target of activity
    - Alice pokes bob, a buyer rates sellers
  - A user's activity influences access control decisions
    - Rating based popularity

Activities in SCS

- No traditional access control can cover all the controls necessary for SCS
- Activity as a key concept for access control

- Why Activity-centric?
  - Multiple kinds of activities (in addition to user's general usage activity against resource) that have to be controlled.
    - User's usage/control activity on user/resource, SCS's service/control activities
  - A user's usage/control activity influences SCS's control decision on own and other users' activities as well as SCS activities.
    - Once Alice invites Bob as a friend, Bob is allowed to see Alice's information
    - If Alice is a friend of Bob and Bob become a friend of Chris, 1) if Chris allows friends of friends to his contents, Alice can access Chris's contents; 2) SCS can recommend Chris and Alice as a friend
    - Buyers' ratings on a seller may collectively used to control the seller's sale activity.

Activity Taxonomy in SCS



Activities in SCS

User's Activities

SCS's Activities

Administrator's Activities

Usage Activities (UA)

.on target resources
.on target users

Control Activities (CA)

.on users
.on resources
.on sessions

Service Activities (SA)

.on target resources
.on target users

Control Activities (CA)

.on users
.on resources
.on sessions

Decision Activities (DA)

.on User's UA, CA
.on SCS's SA, CA

Typical Focus of Access Control

Focus of Discretionary Access Control

Activity Control

User’s Usage Activities

- Usage Activity on Resources
  - Read/view shared comments/photos
  - Typical Focus of Access Control
- Usage Activity on Users
  - Poke, recommend friends

User’s Control Activities

- Control Activity on Resources
  - By changing attributes and policies of resources
  - set a resource as a violent content (attribute), accessible only by direct friends (policy)
  - Parents can set attributes and policies of children’s resources
  - Focus of Discretionary Access Control
- Control Activity on Users
  - By changing user attributes and policies
  - To control activity performed by/against a particular user (self or other related users) without knowing a particular resource
- Control Activity on Sessions
  - By controlling session attributes and policies that are inherited from a user

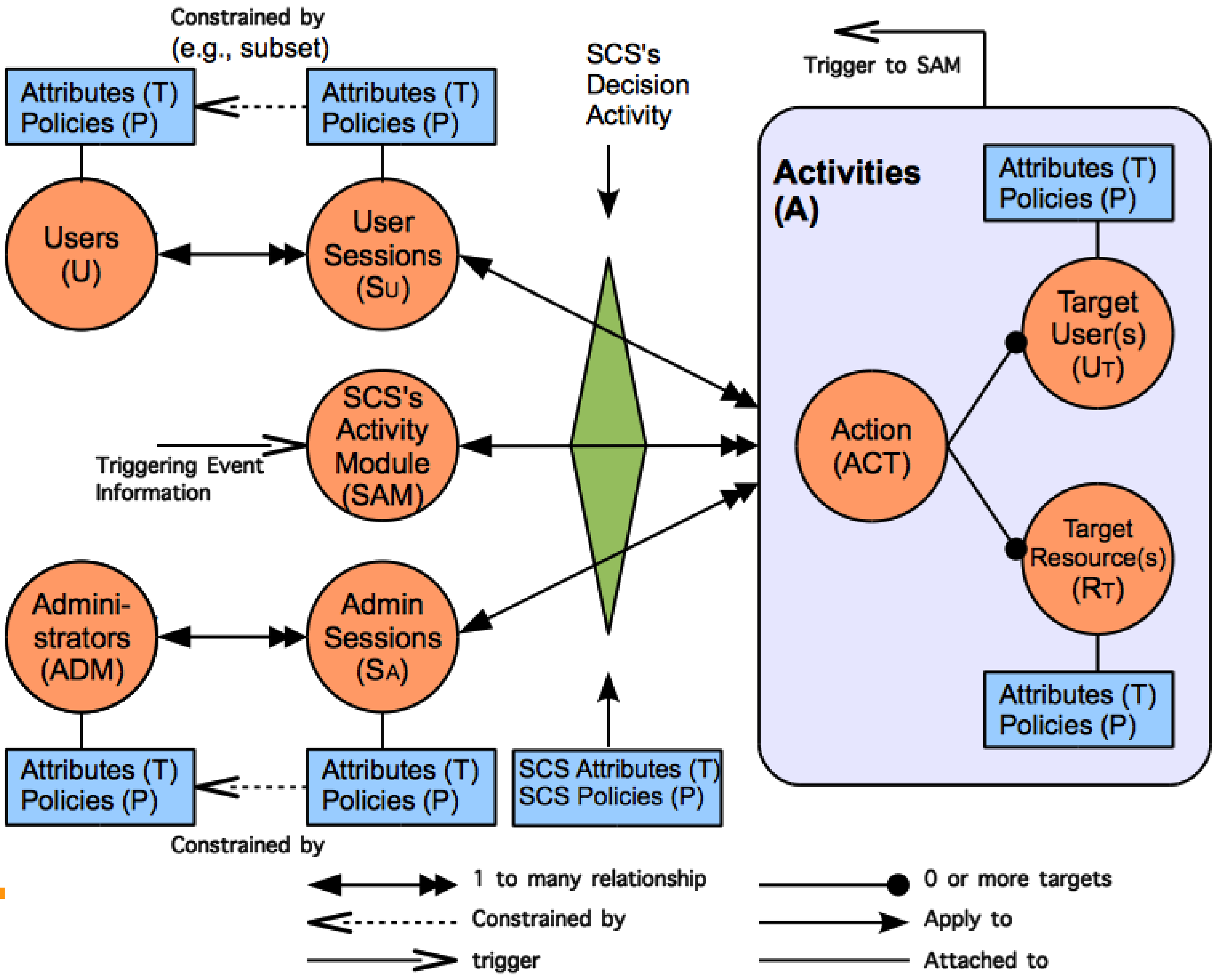
SCS's (Automated) Activities

- Service Activities
  - To promote users' social interactions and information sharing
  - Friends recommendation, friend activity notification, location-based coupons, most-viewed videos
- Control Activities
  - Through managing policies and attributes of users, resources and sessions
  - User rating-based seller trustworthiness or product popularity
- Decision Activities
  - SCS evaluates requests for user's usage and control activities as well as SCS's service and control activities

Activity(-centric Access) Control Framework

- To capture various users and SCS activities and their influences on control decisions
- To support controls on various access/usage and control activities in SCS
- To support personalized user privacy control
- To support automated management of SCS services and controls

ACON Framework



ACON Framework Components

- Users
  - represent a human being who performs activities in an SCS
  - Carry attributes and policies
- Sessions
  - Represent an active user who has logged into the SCS
  - A user can have multiple sessions, but not vise versa
  - Carry attributes and policies that could be different from user attributes and policies

ACON Framework Components (cont)

- Activities
  - User, SCS, SCS administrator’s activities
  - Comprise action, target users, target resources
    - Action
      - An abstract function available in SCS
      - E.g., read, rate, poke, friend-invite, activity notification
    - Target users(’ sessions)
      - Recipients of an action
    - Target Resources
      - Include users’/SCS’s shared contents, user/resource/session policies and attributes

ACON Framework Components (cont)

- SCS's Decision Activity
  - based on the consolidated individual user/resource policies and attributes together w/ SCS policies and attributes
- SCS's Activity Module (SAM)
  - A conceptual abstraction of functions that performs SCS's automated service and control activities
- SCS Administrators
  - Human being w/ a management role

ACON Framework Characteristics

- Policy Individualization
  - A user’s individual policy includes privacy preferences and activity limits
  - Collectively used by SCS for control decision on activities
  - Can be configured by related users
- Separation of user and resource policies
  - User policy allows controls on 1) user activities w/o knowing a particular resource and 2) activities performed against the user w/o knowing a particular resource or the actors
  - E.g., 1) Bart cannot be a friend of Homer’s coworker, 2) Homer doesn’t want to receive violent contents
- User-session distinction
- User relationship independent access control
- SCS’s automated service and control activities

$ACON_{user}$ Model – User Activity Control

- *U, S, ACT, R, T, P, SCS* and *D* (users, sessions, actions, resources, attributes, policies, social computing system and decision predicate, respectively);
- $U_T \subseteq U$ and $R_T \subseteq R$ (target users and target resources, respectively);
- dot notation: we understand *e.T* and *e.P* to respectively denote the set of attributes and set of policies associated with entity *e*;
- *A*, the set of activities is defined as $A \subseteq ACT \times (2^{R_T} \times 2^{U_T} - \emptyset)$;
- Let $A = \{a_1, a_2, ..., a_n\}$, we denote the components of each individual element as $a_i = (a_i.ACT, a_i.R_T, a_i.U_T)$;

$ACON_{user}$ Model – User Activity Control

- $AP\_R_T : A \rightarrow 2^{R_T \times P}$, $AP\_U_T : A \rightarrow 2^{U_T \times P}$, $AT\_R_T : A \rightarrow 2^{R_T \times T}$, $AT\_U_T : A \rightarrow 2^{U_T \times T}$, mappings of activity to a set of target resources and policies, a set of target users and policies, a set of target resources and attributes, and a set of target users and attributes respectively defined as:
  - $AP\_R_T(\{a_1,..,a_n\})=AP\_R_T(\{a_1\}) \cup ... \cup AP\_R_T(\{a_n\})$, $AP\_R_T(\{a_i\})=\{(r_t,p) | r_t \in a_i.R_T, p \in r_t.P\}$
  - $AP\_U_T(\{a_1,..,a_n\})=AP\_U_T(\{a_1\}) \cup ... \cup AP\_U_T(\{a_n\})$, $AP\_U_T(\{a_i\})=\{(u_t,p) | u_t \in a_i.U_T, p \in u_t.P\}$
  - $AT\_R_T(\{a_1,..,a_n\})=AT\_R_T(\{a_1\}) \cup ... \cup AT\_R_T(\{a_n\})$, $AT\_R_T(\{a_i\})= \{(r_t,t) | r_t \in a_i.R_T, t \in r_t.T\}$
  - $AT\_U_T(\{a_1,..,a_n\})=AT\_U_T(\{a_1\}) \cup ... \cup AT\_U_T(\{a_n\})$, $AT\_U_T(\{a_i\})=\{(u_t,t) | u_t \in a_i.U_T, t \in u_t.T\}$;

I·C·S
The Institute for Cyber Security

UTSA

$ACON_{user}$ Model – User Activity Control

- $AP(a)=AP\_R_T(a) \cup AP\_U_T(a)$,
- $AT(a)=AT\_R_T(a) \cup AT\_U_T(a)$;

- $allowed(s,a) \Rightarrow D(s.P,s.T,a,AP(a), AT(a),scs.P,scs.T)$, where $s \in S$ and $a \in A$.

$ACON_{user}$ Model – Session Management

- $user\_sessions : U \rightarrow 2^{S}, session\_users : S \rightarrow U;$
- $user\_added\_sessionT : S \rightarrow 2^{T} , user\_removed\_sessionT : S \rightarrow 2^{T} ;$
- $scs\_added\_sessionT : S \rightarrow 2^{T} , scs\_removed\_sessionT : S \rightarrow 2^{T} , scs\_required\_sessionT : S \rightarrow 2^{T} ;$
- $user\_added\_sessionP : S \rightarrow 2^{P} , user\_removed\_sessionP : S \rightarrow 2^{P} ;$
- $scs\_added\_sessionP : S \rightarrow 2^{P} , scs\_removed\_sessionP : S \rightarrow 2^{P} , scs\_required\_sessionT : S \rightarrow 2^{T} ;$

- $user\_removed\_sessionT(s) \subseteq \{t \in T \mid t \in session\ users(s).T \wedge t \notin scs\_required\_sessionT\ (s)\};$
- $user\_removed\_sessionP(s) \subseteq \{p \in P \mid p \in session\ users(s).P \wedge p \notin scs\ required\_sessionP(s)\};$

$ACON_{user}$ Model – Session Management

- $assignS\_T : S \rightarrow 2^{T}$, $assignS\_P : S \rightarrow 2^{P}$, assignment of attributes and policies to sessions respectively;
- $assignS\_T(s) \subseteq \{t \in T \mid (t \in session\_users(s).T) \vee (t \in user\_added\_sessionT(s)) \vee (t \in scs\_added\_sessionT(s)) \wedge \neg((t \in user\_removed\_sessionT(s)) \vee (t \in scs\_removed\_sessionT(s)))\}$;
- $assignS\_P(s) \subseteq \{p \in P \mid (p \in session\_users(s).P) \vee (p \in user\_added\_sessionP(s)) \vee (p \in scs\_added\_sessionP(s)) \wedge \neg((p \in user\_removed\_sessionP(s)) \vee (p \in scs\_removed\_sessionP(s)))\}$.

Examples

- A buyer can rate a seller only if the buyer bought a product from the seller (SCS.P).
  - *N: a list of users, sellerList :* $S \rightarrow 2^N$
  - $allowed(s, rate, u_t) \Rightarrow u_t \in sellerList(s)$

- A user can recommend a friendship between two friends if they are not a friend to each other(SCS.P).
  - *N: a list of users, friends :* $S \rightarrow 2^N$
  - $allowed(s, f\text{-}recommend, u_t1, u_t2) \Rightarrow (\{u_t1, u_t2\} \in friends(s)) \wedge (u_t2 \notin friends(u_t1)) \wedge (u_t1 \notin friends(u_t2))$

Summary

- Developed activity-centric access control framework for security and privacy in social computing systems.

- Developed initial models for user activity controls and session management.