Delegation of Access Control

Hal Lockhart
February 2011

Disclaimer

The views in this presentation are my own and do not necessarily reflect those of my employer or any other organization.

Summary

- Delegation is a confusing subject
- Many different sorts of scenarios are referred to as delegation
- Many different technology solutions
- I propose a simple structure which appears to encompass all types of delegation
- I will describe design problems and constraints
- I will discuss design tradeoffs
- I will show how different solutions fit into this structure
- I will give examples of multiple ways to implement the same use cases

Delegation Definition

- Not in RFC 2828 or Handbook of Information Security Management (1999)
- Proposed: Giving an Entity the ability to do something that they can’t normally do.
- It is understood that the roles of the several parties are visible to the AC system and part of the criteria for access.
- Delegation is often limited in duration as well as in other ways.

Dispose of a Few Bugaboos

- Ignore exogenous (external) impersonation
  - Jack (authorized) gives Mary (unauthorized) a copy of a secret document
- Ignore endogenous (internal) impersonation
  - Jack tells Mary his password
- Assume dual control increases security over single control
  - Two people have to form a conspiracy to circumvent maker-checker

Delegation examples

- Use cases
  - Assistant approves expense reports
  - Print service reads user's files
  - Veteran delegates access to family members
- Technologies
  - OAuth
  - Kerberos delegation
  - SAML Condition for Delegation Restriction
  - XACML 2.0 Intermediary Subject Category
  - XACML 3.0 per/request policy
  - Web Services Security with Intermediaries

Two Aspects to Delegation

- Dynamic Delegation of Authority (DelAuth)
  - Giving a party the ability to do something they can't normally do by other means than normal administration
- Dynamic Delegation of Action (DelAct)
  - Giving a party the ability to do something they can't normally do by virtue of the fact that it is being done as a part of processing a specific request by another party

Temporary Definition

- For the purposes of this talk Authority means: all administratively modifiable information which contributes to an access control decision
  - Includes: attribute values, policies, ACLs, Roles, permissions, delegation tokens
  - Does not include: hardware, code

Dynamic Delegation of Authority

- Access Control always involves prior delegation of authority (create policies, update attributes, etc.)
- Delegation of Authority can be:
  - Not allowed
  - CanDoCanDel – “Anything you can do, you can delegate”
  - Constrained – Arbitrarily limited e.g. XACML 3

Dynamic Delegation of Authority

- May be performed in one of two ways
  - In advance – usually covers a class of situations
  - Just in time – covers the immediate request
- Secure processing requires
  - At delegation time
    - Statement of scope bound to Trusted Issuer
    - Identification & Authentication of Trusted Issuer
    - Identification of Delegate
  - At access time
    - Access to above information
    - Authentication of Delegate
  - May be done at either time
    - Determination that scope is valid for Issuer

Dynamic Delegation of Action

- Access is allowed specifically because it is all or part of a request by another party
- Secure processing requires
  - Policy model able to account for multiple parties
  - Static or dynamic Authority
  - Identification & Authentication of parties, bound to request

Design Issues

- Client limitations
  - Ability to implement new protocol
  - Ability to perform crypto operations
  - Secure access to keys
- For Delegation of Authority
  - Expressing and Evaluating Delegation Limits
    - Can-Do-Can-Del is one way to solve this
    - Policy comparison may be difficult
  - Expressing scope of delegation
    - Standardization – syntax & semantics
    - Determining if request is in scope

Delegation Tradeoffs

- Delegation of Authority
  - More flexible, more ad hoc, less efficient
  - Use for less common or highly variable situations
  - Use when only a small % of population has requirement
- Delegation of Action
  - Less flexible, more rigid, more efficient
  - Cover common, complex cases
  - Consider combining both types for the most complex requirements

Technologies

- OAuth 3 Leg
- OAuth 2 Leg
- Kerberos delegation
- SAML Condition for Delegation Restriction
- XACML 2.0 Intermediary Subject Category
- XACML 3.0 per/request policy

OAuth 3 Leg

- DelAuth & DelAct
- User requests print service print file on fileserver
- Print service Authenticates to Authorization Service and gets session key
- User is redirected to Authorization Service to Authenticate and approve access
- Authorization service provides token specifying access, bound to print service
- Print service presents token to file server

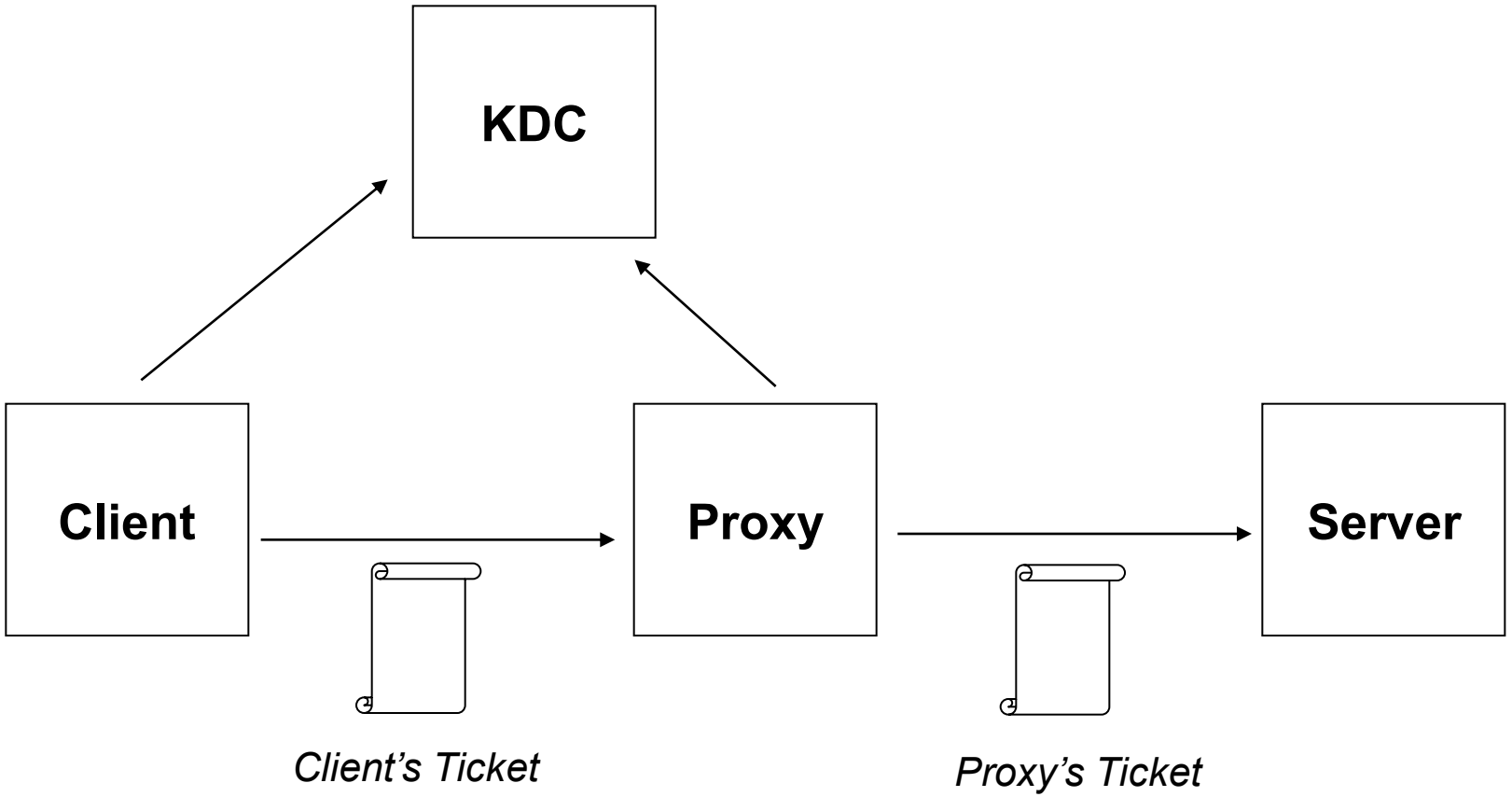
OAuth 3 Leg

- Token represents DelAuth
- Just in time delegation model
- Interaction with user provides DelAct
- Scope expression unspecified
  - However it is produced and consumed by resource server
  - Interoperability not a problem initially
  - Implies use of Can-Do-Can-Del
- Issuer can be strongly authenticated
- Request weakly bound to Intermediary

OAuth 2 Leg

- No user interaction
- Supports offline usecases
- Standardization in process (OAuth 2.0)
- Scope format still unspecified
- Scope limits also unspecified
- Delegation in advance for multiple requests

Kerberos Delegation



Classic Kerberos Delegation

- Defined with rest of Kerberos in RFC 4120
- Two methods
  - Allow server to act on behalf of client to 2nd server
  - Initiated by client
  - Proxying
    - KDC issues service ticket for proxy
    - Client can subset authorized capabilities
    - Ultimate server can refuse to accept proxy
  - Forwarding
    - KDC issues TGT on behalf of client for proxy
    - Proxy can get service ticket to any server
- Not much used in practice – security risks

Service for User (S4U) Delegation

- Invented by Microsoft
- S4U to Self (S4U2Self)
  - Proxy requests service ticket to itself for any user
  - User need not authenticate or be present
  - Convenient way to get user AuthN data in std format
  - Allows access by client using non-Kerberos Authentication
- S4U to Proxy (S4U2Proxy)
  - Proxy provides client TGT and requests proxy service ticket
  - Just like classic proxying, except initiated by Proxy
  - When combined with SU42Self, can act as proxy for any user
  - Significant security risk from attack on Proxy

Kerberos Delegation

- Proxying & S4U2Proxy - Delegation of Authority
- Only difference is who initiates
- Scope expression unspecified
- Scope validity unspecified (Can-Do-Can-Del may have been the intent)
- Strong authentication of trusted issuer (KDC)
- Identification of delegate via Kerberos tickets
- Strong authentication based on Kerberos mechanisms

SAML Condition for Delegation Restriction

- Profile allows addition of <Delegate> to <Conditions>
- <Delegate> contains <Subject> information recording the intermediaries
- Implementation built by Internet2
  - Based on ECP + WSS/TLS
  - SP enforces policy considering intermediary identities

SAML Condition for Delegation Restriction

- Delegation of Action
- Looks like Kerberos and OAuth, but Subjects are not who is authorized, rather who has been in chain
- Intermediary-aware policy model – proprietary
- Identities bound indirectly to request via Assertion
- Strength of binding depends on Confirmation method used
- In turn depends on ability of entities to do crypto, etc.

Delegation with XACML 2.0

- Use of Intermediary Subject Category
  - Print Format Service can read any file a user wants printed, but not otherwise
  - Access Subject + Intermediary Subject
- Delegation by modifying attributes
  - User can enable family member's access
  - Policy protects subject repository
- Policies protecting each policy repository

XACML 2.0 Intermediary Subject

- Delegation of Action
- Requires use of protocol which can record participating Intermediaries
- For Example, WSS with counter signatures
  - Originator signs message
  - Intermediary adds signature over
  - WSS Security Token Reference – wsse:Usage attribute
- Policy based on properties of Access Subject and Intermediary Subject

XACML 3.0 Administration/Delegation

- Two primary use cases
  - "HR-Admins can create policies concerning the Payroll servers"
  - "Jack can approve expenses while Mary is on vacation"
- Backward compatible
- Defined as an optional Profile
- Policies can contain Issuer
- Policies can be Access or Admin
- Admin policies enable policy creation

- New Function – access-permitted(Category, Attributes)
  - Implements generalization of Can-Do-Can-Del

Policy Evaluation

1. Select potentially applicable policies by Target matching
2. For each Policy evaluate Rules and combine
   - Target Match
   - Evaluate condition
   - Return Effect and associated Obligations
3. For each Policy Set combine policy results
4. Return Effect and Obligations

Policy Evaluation with Admin Policies

1. Select potentially applicable policies by Target matching
2. For each Policy evaluate Rules and combine
   - Target Match
   - Evaluate condition
   - Return Effect and associated Obligations
3. For every un-trusted policy
   - Find an applicable Admin policy which authorizes the Issuer
   - Repeat until a chain to a trusted policy is found
   - Discard unauthorized policies
4. For each Policy Set combine policy results
5. Return Effect and Obligations

XACML 3.0 per/request policy

- Delegation of Authority
- Administrative policy allows user to create certain policies at runtime
- At time of request, user provides signed, enabling policy
- XACML/SAML Decision Request can carry policies to be added top top level Policy Set for this decision only
- Avoids scope comparison issue by comparing policies to Request Context, not to each other
- Optionally can use access-permitted()

Design Patterns

- Print Service
  - Delegation of Authority
  - Delegation of Action
- Family Members
  - Delegation of Authority via repository
- Vacation approvals
  - Four different approaches

Print service reads user's files

- Scheme P1 – fixed policy (DelAct)
  - Fileserver policy says print service can read any file requested by a party able to read file
  - User sends signed request to print server
  - Print server requests file access
    - Includes print request
    - Signs over both requests
- Scheme P2 – see OAuth 3 step (DelAuth)

Veteran delegates access to family members

- Delegation of Authority
- Policy says family members allowed to access veterans info
- Repository access allows Vet to designate others as family
- Request is checked to see if requestor is Vet owning data or family member of same Vet

Assistant Approves Expenses

- Scheme E1 – Policy is assistants can always approve expenses – nothing dynamic
- Scheme E2 – Boss indicates state of ”away”
  - Alternate approver defined for all approvers
  - Policy allows alternate to approve
- Scheme E3 – Boss indicates identity of alternate approver
  - Policy allows alternate to approve
- Scheme E4 – OAuth 2 Leg used to get AuthN Token
  - Alternate Approver presents token with request

References - 1

- OAuth 1.0 (RFC 5849)
  - http://tools.ietf.org/html/rfc5849
- OAuth 2.0 Latest Draft (11)
  - http://tools.ietf.org/html/draft-ietf-oauth-v2-11
- Kerberos
  - RFC 4120
    - http://www.ietf.org/rfc/rfc4120.txt
  - Useful Kerberos Documents
    - http://www.kerberos.org/docs/links.html
- Web Services Security
  - http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf

References - 2

- SAML Condition for Delegation Restriction
  - Specification
    - http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-delegation-cs-01.pdf
  - Internet2 Implementation
    - https://spaces.internet2.edu/display/ShibuPortal/Home
    - https://spaces.internet2.edu/display/ShibuPortal/Solution+Proposal
    - https://spaces.internet2.edu/display/SHIB2/NativeSPPolicyRule#NativeSPPolicyRule-DelegationRule%28Version2.2andAbove%29
- XACML
  - XACML 3.0 core
    - http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-cd-03-en.pdf
  - XACML 3.0 Administration Profile
    - http://docs.oasis-open.org/xacml/3.0/xacml-3.0-administration-v1-spec-cs-01-en.pdf