Extending User Controlled Security Domain with the TPM/TCG in Grid based Virtual Collaborative Environment

Yuri Demchenko <demch@science.uva.nl>
(in collaboration with Leon Gommans, Cees de Laat)
System and Network Engineering Group
University of Amsterdam

COLSEC2007 Workshop
22 May 2007, Orlando, Florida, US
Outline

- Security and trust in Collaborative applications
  - User and Service Provider vs System and Data
- Virtual Workspace Service in Grids
- Trusted Computing Platform and Trusted Platform Module
- User Controlled Virtual Workspace (VWSS-UC) organisation
- AuthZ session management and AuthZ ticket format
- Summary and future development
- Discussion – Vision for use of TPM/TCG technology
Virtual Laboratory (VL) as a business Collaborative Environment

- Implementing Utility Computing paradigm
- Can a VL provider offer a trusted experiment environment from the competitor’s point of view
  - Extreme usecase: *Will Pepsi Company trust to do analysis on the Coca-Cola VL facility?*
  - Common sense: *Remote System can be trusted as much as the system administrator is trusted*

Content providers (music, movie)

- Content played at the user PC/player should be protected from copying or useable during the service contract

Service/Resource providers (service on demand)

- Enforce use of on-demand provided resource and Policy Obligations
Different sides of the Security and Trust

• Modern paradigm of remote distributed services and digital content providing makes security and trust relations between User and Provider more complex

• User and Service Provider – two actors concerned with own Data/Content security and each other System/Platform trustworthiness

• Two other aspects of security/trust
  ◆ Data stored vs Data accessed/processed
  ◆ System Idle vs Active (running User session)

• Think about real life analogy: *Diplomatic/President’s visit*

\[\text{User} \quad \text{Provider} \]
\[\text{System} \quad \text{System} \]
\[\text{Data} \quad \text{Data} \]

\[\text{Trust(worthiness)} \quad \text{Security}\]
Background research and target projects

- **Distributed Authorisation infrastructure for Grid based Collaborative applications**
  - GAAA-AuthZ Architecture and implementation (Collaboratory.nl, VL-e)
    - AuthZ session/ticket for AuthZ service performance optimisation (@ CTS2006)
    - Domain based hierarchical resource management (GAAA-DM)
- **Distributed multidomain Authorisation service for network on-demand services and OLPP**
  - EU Project PHOSPHORUS and NL national project RoN GP-NG
    - Extended AuthZ session context and trust management in multidomain scenario
- **Open Grid Forum (OGF) Grid and Virtualisation WG**
  (https://forge.fridforum.org/sf/projects/gridvirt-wg)
  - Security model for virtualised Grid applications
Domain based Resource management in GCE

Full Resource URI/ID –
**CNL:Facility:VirtualLab:Experiment:InstrModel**

Full User Session context –
**Facility < Virtual Lab < Experiment < Experiment Session < Collaborative Session**

Implements RBAC3 model + Experiment AuthZ session management
Uses XACML RBAC profile and XACML v3.0 administrative policy profile
GCE/VL Security infrastructure requirements

- Dynamically create user/application workspaces (together with related security services)
- Dynamically create user accounts and handle different/multiple user identities and credentials
  - (Securely) associate multiple administrative and trust domains (e.g., by means of the Virtual Organisation (VO) or other Identity federation forms)
  - Negotiate and handle multiple security and access control policies (for both resource provisioning and access stages)
- Manage session based user and application security context
- Allow for user rights/roles delegation, including delegated hierarchical policies administration
- Allow for binding the whole chain of trust in dynamic collaborative sessions to the VL facilities/platform root of trust or to User credentials
Trusted Collaborative Environment components

- Trusted Computing Platform
- Virtual Workspace Service (VWSS)
- Application/Resource (dynamic) access control service

- 3 layer User Controlled VWSS (VWSS-UC)
Globus Toolkit Virtual Workspace Service (VWSS)

  - Comprises of the Workspace Factory Service (WFS) and the Workspace service
  - Built as VM/Xen-based virtual environment
- Current security model provides only WFS access control using basic GT4-AuthZ service
  - Can use also (trusted) secure storage for user pre-configured VM images
  - Relies on the Grid service provider trust
TCG Trusted Computing Platform

Promoted by the Trusted Computing Group (TCG)
- Basis for building and managing controlled secure environment for running applications and processing (protected) content
  - https://www.trustedcomputinggroup.org/home
- Standards for trusted network, client, server and mobile agent
- TMP software stack (TSS) defines API’s for remote access, Identity Mngnt, PKI, Secure e-mail, file/folder encryption, etc.

TCG components
- **Trusted Platform Module (TPM)**
- “Curtained memory” in the CPU
- Security kernel in the OS and security kernel in each application
- Back-end infrastructure of online security servers maintained by hardware and software vendors

**Trusted Network Connect (TNC)** – to enforce security policies before and after endpoints or clients connect to multi-vendor environment
Trusted Platform Module (TPM)

Chip built-in into the computer system or a smartcard chip

- Can be considered as a platform tied “root-of-trust” and used for trusted platform registration and integrity assurance

Provides a number of hardware-based cryptographic functions

- **Asymmetric key functions** for on-chip key pair generation using hardware random key generation; private key signatures; public key encryption and private key decryption
- An **Endorsement key** that can be used by a platform owner to establish that identity keys were generated in a TPM, without disclosing its identity
- **Direct Anonymous Attestation (DAA)** that securely communicates information about the static or dynamic platform configuration, which is internally stored in TPM in the form of hashed values (based on Zero-knowledge cryptography)
- Monotonic counter and the tick counter to enable **transaction timing and sequencing**
- Protection of communication between two TPM’s
- Secure key/data backup to another TPM
User-controlled Virtual Workspace Service (VWSS-UC) – 3 layer model

- Trust Anchors: T0 (TPM) – TA1 (VM/VWSS) – TA2 (Application) – TA# (User)
- VWSS session and Application AuthZ sessions
VWSS-UC – Implementation Suggestions

TPM Enabled computer platform
  - http://www.tonymcfadden.net/tpmvendors.html

Growing number of TCG/TPM oriented projects to develop TMP oriented firmware and middleware
  - Daonity (HP), OpenTC (EU), number of nationally funded projects in Germany, Czech Republic, associated research in EGEE and UvA

Xen v3.0 has already so-called Virtual TPM module
  - http://www.cl.cam.ac.uk/Research/SRG/netos/xen/readmes/user

Grid Virtual Workspace Service (VWSS) – GT4 candidate component
  - http://workspace.globus.org/

GAAA-AuthZ Authorisation session management supported by GAAAPI
  - Proprietary and SAML based AuthZ ticket formats
AuthZ Session management in GAAA-AuthZ

- AuthZ session is a part of the generic AAA-AuthZ functionality
- Session can be started only by an authorised Subject/Role
  - Session can be joined by other less privileged users
  - Session permissions/credentials can be delegated to (subordinate) subjects
- Session context includes Request/Decision information and may include any other environment or process data/information
  - AuthZ Session context is communicated in a form of extended AuthZ Assertion or AuthZ ticket
  - SessionID is included into AuthzTicket together with other AuthZ Ctx information
  - Signed AuthzTicket is cached by PEP (Policy Enforcement Point) or PDP (Policy Decision Point)
- If session is terminated, cached AuthzTicket is deleted
  - Note: AuthzTicket revocation should be done globally for the AuthZ trust domain
AuthZ session Tickets/Tokens handling in AuthZ system

- AuthzTicket is issued by PDP and may be issued by PEP
- AuthzTicket must be signed
- AuthzTicket contains all necessary information to make local PEP-Triage Request verification
- When using AuthzTokens, AuthzTickets must be cached; Resolution mechanism from token to ticket must be provided
AuthZ ticket/assertion for extended security context management – Data model (1) - Top elements

Required functionality to support multidomain provisioning scenarios
- Allows easy mapping to SAML and XACML related elements
- Allows multiple Attributes format (semantics, namespaces)
- Establish and maintain Trust relations between domains
  - Including Delegation
- Ensure Integrity of the AuthZ decision
  - Keeps AuthN/AuthZ context
  - Allow Obligated Decisions (e.g. XACML)
- Confidentiality
  - Creates a basis for user-controlled Secure session
AuthZ ticket Data model (2) - Mandatory elements

- TicketID attribute
- Decisions element and ResourceID attribute
- Conditions Element and validity attributes
- Extensible element ConditionAuthzSession
  - Any AuthZ session related data
AuthZ ticket Data model (3) – Subject and Delegation elements

- Subject element to keep AuthN security context and Subject Attributes
- Delegation element to allow permissions/AuthZ decision delegation to other Subjects or groups/community
AuthZ ticket main elements

**<Decision>** element - holds the PDP AuthZ decision bound to the requested resource or service expressed as the ResourceID attribute.

**<Conditions>** element - specifies the validity constraints for the ticket, including validity time and AuthZ session identification and additionally context

  **<ConditionAuthzSession>** (extendable) - holds AuthZ session context

**<Subject>** complex element - contains all information related to the authenticated Subject who obtained permission to do the actions

  **<Role>** - holds subject’s capabilities

  **<SubjectConfirmationData>** - typically holds AuthN context

  **<SubjectContext>** (extendable) - provides additional security or session related information, e.g. Subject’s VO, project, or federation.

**<Resources>/<Resource>** - contains resources list, access to which is granted by the ticket

**<Actions>/<Action>** complex element - contains actions which are permitted for the Subject or its delegates

**<Delegation>** element – defines who the permission and/or capability are delegated to: another **DelegationSubjects** or **DelegationCommunity**

  • attributes define restriction on type and depth of delegation

**<Obligations>/<Obligation>** element - holds obligations that PEP/Resource should perform in conjunction with the current PDP decision.
AuthZ ticket format (proprietary) for extended security context management – 3-10KB

```xml
TicketID="cba06d1a9df148cf4200ef8f3e4fd2b3">
  <AAA:Decision ResourceID="http://resources.collaboratory.nl/Philips_XPS1">Permit</AAA:Decision>
    <!-- SAML mapping: <AuthorizationDecisionStatement Decision="*" Resource="*"> -->
  <AAA:Actions>
    <AAA:Action>cnl:actions:CtrlInstr</AAA:Action> <!-- SAML mapping: <Action> -->
    <AAA:Action>cnl:actions:CtrlExper</AAA:Action>
  </AAA:Actions>
  <AAA:Subject Id="subject">
    <AAA:SubjectID>WHO740@users.collaboratory.nl</AAA:SubjectID> <!-- SAML mapping: <Subject>/NameIdentifier> -->
    <AAA:SubjectConfirmationData>IGHllmava8YQomTg9Ege9JRN1ld84AggaDkOb5WW4U=</AAA:SubjectConfirmationData>
      <!-- SAML mapping: EXTENDED <SubjectConfirmationData/> -->
    <AAA:Role>analyst</AAA:Role> <!-- SAML mapping: <Evidence>/Assertion>/AttributeStatement>/Assertion>/Attribute>/AttributeValue> -->
    <AAA:SubjectContext>CNL2-XPS1-2005-02-02</AAA:SubjectContext>
      <!-- SAML mapping: <Evidence>/Assertion>/AttributeStatement>/Assertion>/Attribute>/AttributeValue> -->
  </AAA:Subject>
  <AAA:Delegation MaxDelegationDepth="3" restriction="subjects">
    <!-- SAML mapping: LIMITED <AudienceRestrictionCondition> (SAML1.1), or <ProxyRestriction>/Audience> (SAML2.0) -->
    <AAA:DelegationSubjects> <AAA:SubjectID>team-member-2</AAA:SubjectID> </AAA:DelegationSubjects>
  </AAA:Delegation>
    <!-- SAML mapping: <Conditions NotBefore="*" NotOnOrAfter="*"/> -->
  </AAA:Conditions>
  <AAA:Obligations>
    <AAA:Obligation>put-policy-obligation(2)-here</AAA:Obligation> <!-- SAML EXTENDED: <Advice>/PolicyObligation> -->
    <AAA:Obligation>put-policy-obligation(1)-here</AAA:Obligation>
  </AAA:Obligations>
</AAA:AuthzTicket>
```

<ds:Signature> <ds:SignedInfo/> <ds:SignatureValue>e4E27kJwEXoVdnXIBpGvpaBGvY71Nypos...</ds:SignatureValue></ds:Signature>
AuthzToken example – 293 bytes

```xml
<AAA:AuthzToken TokenID="c24d2c7dba476041b7853e63689193ad">
  <AAA:TokenValue>
    0IZt9WsJT6an+tIxhhTPtiztDpZ+iynx7K7X2Cxd2iBwCUTQ0n61Szv81DKl1Wsq75IsHfusnm56zT3fhKU1zEUsob7p6oMLM7hb42+vjfvNeJu2roknhIDzruMrr6hMDsIfaotUREpu7QCT0sADm9IfX89Et55EkSE9oE9qBD8=
  </AAA:TokenValue>
</AAA:AuthzToken>
```

AuthzToken is constructed of the AuthzTicket TicketID and SignatureValue. AuthzToken use suggests caching AuthzTicket's. AuthzToken can be used as cookie in Web/portal based applications.
Conclusion and Future developments

- TCG Trusted Computing platform allows for further extension of the user/provider controlled/trusted security domain
- Virtualised Workspace and dynamically provisioned resources can use TP security anchor(s) to provide User-Resource end-to-end trust
  - Proposed AuthZ session model and AuthZ ticket format are aimed for this

- More formal definition of the proposed model is needed
  - Contribute to the OGF Virtualisation WG use cases and security model
  - Propose AuthZ session management framework to OGSA-AUTHZ

- Dynamic Trust management in multidomain Complex Resource Provisioning (CRP) for TPM enabled resources

- Implementation: Add TPM support to GT4-VWSS and AuthZ support to Grid oriented AuthZ frameworks EGEE gJAF and GT4-AuthZ
Discussion

Vision for wider use of TPM/TCG technology in Collaborative applications?
Additional information

- AuthZ service components in GAAA-AuthZ and gJAF/GT4-AuthZ
GAAA/AuthZ/GAAAPI components to support dynamic security context management (1)

- GAAAPI is a collection of components to support PEP and PDP interaction, implemented in Java
- Needs Trust Anchor configuration in a distributed multidomain infrastructure
GAAAPI components to support dynamic security context management (2)

- Context Handler (CtxHandler) that calls to a namespace resolver (NS Resolver) and attribute resolver (AttrResolver), which in its own can call to external CVS or Attribute Authority Service (AAS) to validate presented attributes or obtain new ones.
- Triage and Cache to provide an initial evaluation of the request, including the validity of the provided credentials:
  - Used for handling AuthZ tickets/tokens, and also for AuthZ session management by evaluating service requests versus the provided AuthZ ticket/token claims.
- Ticket Authority (TickAuth) generates and validates AuthZ tickets or tokens on the requests from PEP or PDP:
  - to support AuthZ session, tickets are cached by TickAuth directly or by PEP/PDP.
- Policy Information Point (PIP) that provides resolution and call-outs to related authoritative Policy Authority Points (PAP).
gJAF – Proposed Extensions for AuthZ Session Management