

# **Proposal for Secure Access Control for OGC services WFS and WMS**

Andreas Matheus

Technische Universität München

D-85747 Garching, Germany

matheus@in.tum.de

# Motivation

-The 'moneymaking' geodata is not freely available-

- ◆ OGC services provide interoperability for accessing geodata
- ◆ But, without secure access control, no use
- ◆ Poll at the German fair Intergeo (10-2003) showed that
  - Major GIS companies provide OGC services
  - But, same companies also implement proprietary access control and security mechanisms
- ◆ No interoperability on access control level without OGC 'intervention'

# Requirements for access control

-Poll at German fair Intergeo and research project-

- ◆ Fine grained: Assuming that the geodata is somehow structured
- ◆ Service result is a set of features (view), where the features are of specific type
- ◆ Security: Ensure confidentiality about features
  - Confidentiality of result features must be ensured
- ◆ Spatial: Features, inside of defined areas (box, polygon) may be restricted

# Access Control Model

## -View based approach-

- ◆ A service operation returns a -dynamically created- view (set of features)
- ◆ A view contains a set of spatial features
  - Has a format, e.g. GIF, SVG, GML2
  - May have style, depending on the format
- ◆ Each feature of a view
  - Is an instance of a particular feature type
  - Has spatial and non-spatial properties
  - Can be confidential

# Authorizing Subjects

## -Base set of information-

- ◆ Based on subject 'local' identity properties such as Name, Email
- ◆ Based on subject 'remote' identity using an X.509 certificate
- ◆ Based on the network (IP) address of the client in numeric notation
- ◆ Based on the symbolic name of the client in dot notation
- ◆ Request can be characterized by triple <Subject ID, IP address, Symbolic name>

# Authorizing Subjects

## -Use of patterns (wildcards)-

- ◆ Build subject groups, using explicit listing
  - Group = {Joe, Alice}
  - Group = {email.name=Joe@yahoo.com}
- ◆ Build groups of network addresses
  - Group = {10.10.2.1, 10.\*.\*.\*}
- ◆ Build groups of symbolic names
  - Group = {\*.yahoo.com, client.\*.de}
- ◆ Example authorization <Alice,10.10.2.1,\*>
  - Subject Alice is authorized if connecting from IP 10.10.2.1, regardless of symbolic name

# Object side restrictions

## -The base set-

### ◆ View may have

- Format restrictions: Vector or binary formats
- Style restrictions: E.g. black&white or color

### ◆ Features may have

- Type restrictions: Each feature is of particular type
- Instance restrictions: Each feature is unique
- Spatial restrictions: Some feature have geometry
- Access restrictions: Read, Create, Delete, Modify
- Confidentiality restrictions:
  - ◆ Communication must be confidential
  - ◆ User must use X.509 certificate for prove of identity

# Object side authorization

-Propagation of non spatial authorizations-

- ◆ An authorization can be positive (+) or negative (-) or nothing ( $\epsilon$ )
- ◆ An authorization can apply to
  - One feature or type
  - Authorization upon one feature type applies to all feature instances of that type
  - The authorization can be
    - ◆ Non-recursive: Applies to the actual feature or type only
    - ◆ Recursive: Applies to actual and all descending features
    - ◆ Inherited: Applies to all inherited feature types
- ◆ Explicit overrides implicit authorization

# Object side authorization

## -Propagation of spatial authorizations-

- ◆ A spatial authorization can be positive (+) or negative (-) or nothing ( $\epsilon$ )
- ◆ A spatial authorization applies to 2D features
- ◆ 0D and 1D features take authorization of enclosing area
- ◆ A spatial authorization can apply
  - Non-recursively: Applies to the actual area only
  - Recursively: Applies to actual and all inside areas
- ◆ Explicit overrides implicit authorization

# Authorization engine

-Grant/deny a request upon input parameters-

- ◆ View format, style restriction
  - WMS getMap request: ...,format=GIF, style=black,...
- ◆ Feature type restriction
  - WFS getFeature request: ...,typename=aType,...
- ◆ Feature instance restriction
  - WFS getFeature request: ...,featureID=id4711,...
- ◆ Spatial restriction
  - WMS getMap request: ...,BBOX=0,0,1,1,...
- ◆ Access restriction
  - WFS getFeature request: Implies a read access

# Authorization engine

-Grant/deny a request upon resulting view-

## ◆ Spatial restriction

- WFS getFeature request: ...,featureID=id4711,...
  - ◆ Here, the BBOX (if any) of the result must be calculated (or taken) from the resulting view

## ◆ Instance restriction

- WMS getMap request: ...,BBOX=0,0,1,1,...
  - ◆ Here, XML document of features building the image map must be available as meta information.
  - ◆ Fine grained authorization is possible from this document
- WFS getFeature request with filter on properties
  - ◆ Here, feature instance information come from result view

# Authorization engine

## -Guarantee confidentiality restrictions-

- ◆ Confidentiality is based on X.509 certificates
  - Must be issued by a trusted authority
- ◆ Confidential transmission from service to client
  - Confidential feature instances must be encrypted
  - The subject's public key is required
- ◆ User Authentication method
  - If resulting view contains confidential features, user must use X.509 certificate to prove identity

# Interoperability issues

## -Use of SAOP and XACML-

### ◆ XACML allows

- Interoperable specification of restrictions
- Use of Xpath expressions for fine grained access control on GML documents and application schemas
- Use of standard tools for deriving authorization decisions

### ◆ SOAP provides processing of meta information

- Processing of header elements must be agreed on
- Structure of header elements must be agreed on

### ◆ XML Encryption enables

- Element encryption of confidential feature instances

# Implementation issues

## -Service extension vs. façade to services-

- ◆ Authorization engine implemented as façade
  - In general, processing of service result is essential
  - For WMS, no instance authorization possible for binary formats
- ◆ Authorization engine is inside OGC services
  - Service specification must be extended
  - Use of HTTP-Get/Post remains unchanged!?
  - Additional capabilities (authorization) is available for SOAP binding only
  - Use of SOAP: Specification must describe the names, structure and processing of header elements

# Upcoming work

-In co-operation with OGC, if favored-

- ◆ Short term (until April 2004)
  - Preparation of discussion paper about this topic before 3 week rule of next TC in April 2004
- ◆ Long term (until end of 2004)
  - Implementation of the access control processor as façade or extension to WFS and WMS
  - Student thesis are been/will be worked on
- ◆ Evaluation upon basic WFS prototype of Thorsten Kunkel ([tk@thorsten-kunkel.de](mailto:tk@thorsten-kunkel.de))
- ◆ Any (full) WFS, WMS implementations available for evaluation?