Технологии и средства межорганизационной Аутентификации и Авторизации

RELARN2004
3 июня, 2004

Yuri Demchenko, University of Amsterdam
<demch@science.uva.nl>

Содержание

- Особенности Аутентификации (AuthN) и Авторизации (AuthZ) в научных и образовательных сетях
- Основные технологии
  - Особенности сервисов AuthN/AuthZ в Grid
- Существующие системы распределенной аутентификации и авторизации
  - Инфраструкутра распределенной аутентификации A-Select
- Примеры проектов (Collaboratory.nl, EGEE)
- Справочная информация
  - Архитектура безопасности на основе XML

AuthN/AuthZ в научных и образовательных сетях

- Доступ к многосайтовым Веб/Интернет ресурсам
  - Перенаправление + cookie (SSO)
- Межуниверситетские ресурсы и доступ к внешним ресурсам или предоставление доступа для внешних пользователей
  - Например, библиотечные каталоги или научные БД
- Распределенные университетские кампусы и дистанционное обучение
- Грид-центры и Грид-приложения

- Общие характеристики/проблемы
  - Различные административные домены и домены безопасности
  - Единый доступ (SSO – Single Sign On) и множество паролей
  - Разделение идентификации/аутентификации и управления доступом

Особенности AuthN/AuthZ в Grid

- Операционые особенности
  - Различные административные домены и домены безопасности
  - Задачи могут быть продолжительные, изменяемые и мобильные
    - Требование к аутентичности, целостности и доверию
  - Динамические ресурсы
  - Использование прокси-удостоверений (proxy-credentials) и делегирование
- Виртуальные организации (ВО)
  - Сервисы безопасности как и в реальных организациях (РО), но создаваемые динамически
  - ВО как стандартный Грид-сервис
    - Динамически создаваемые и ликвидируемые
  - Обычно имеет место членство во многих ВО
  - Согласованность политики безопасности между ВО и РО

Современная архитектура сервисов AuthN и AuthZ

Требования к современной архитектуре AuthN/Z

- Разделение сервисов аутентификации (AuthN) и авторизации (AuthZ)
  - Аутентификация в «домашней»/«родной» организации
  - Авторизация ресурсом
- Конфиденциальность, приватность и анонимность
- Управление доступом на основе ролей (RBAC – Role Based Access Control) и использование политики безопасности

Проблемы

- Множество логинов/паролей – на каждый ресурс/сайт
- Ограничение одним доменом безопасности или множество сертификатов открытых ключей
- Сложность частичной динамической делегации полномочий

Базовые технологии

- LDAP директории и метадиректории для храненения данных о пользователях
- Собственные системы авторизации

Использование LDAP в сервисах AuthN/AuthZ

Структуры персональных данных в LDAP

- Person (RFC2256), organisationalPerson (RFC2256), InetOrgPerson (RFC2798)
- EduPerson – расширение для образовательных организаций

Основные атрибуты Person objectClass

- sn/surName
- cn/commonName
- givenName
- uid, displayName
- **userPassword**
- x500uniqueIdentifier
- userCertificate
- userSMIMECertificate
- userPKCS12
- postalAddress
- o/organizationName
- ou/organizationalUnitName
- st/stateOrProvinceName
- l/localityName
- c/country
- title,employeeType
- mail
- photo

Дополнительные атрибуты EduPerson (всего 43)

- eduPersonAffiliation
- eduPersonNickname
- eduPersonOrgDN
- eduPersonOrgUnitDN
- eduPersonPrimaryAffiliation
- eduPersonPrincipalName
- eduPersonEntitlement
- eduPersonPrimaryOrgUnitDN

Безопасность приложений на основе XML и традиционная модель сетевой безопасноти

Традиционная модель сетевой безопасности (ISO7498-2):

- Host-to-host или point-to-point безопасность
- Ориентированная на архитектуру клиент/сервер
- Ориентированные на соединении (connection-oriented) и без соединения (connectionless)
- В общем случае единый доверительный домен (на основе PKI)

Безопасность приложений на основе XML

- Безопасность между конечными точками или приложениями (end-to-end)
- Ориентированная на документ (или семантический обьект)
  - Мандаты и маркеры безопасности могут быть ассоциированы с документом или сообщением или их частью
- Существующие технологии WS-Security обеспечивают безопасность между разными административными доменами и доменами безопасности
- Позволяет создавать динамические и виртуальные ассоциации безопасности

Управление доступом на основе ролей

RBAC – Role Based Access Control

- Роль описывает функцию и определяет права/привилегии
- Права определяют доступ к ресурсу в определенном режиме

Преимущества RBAC

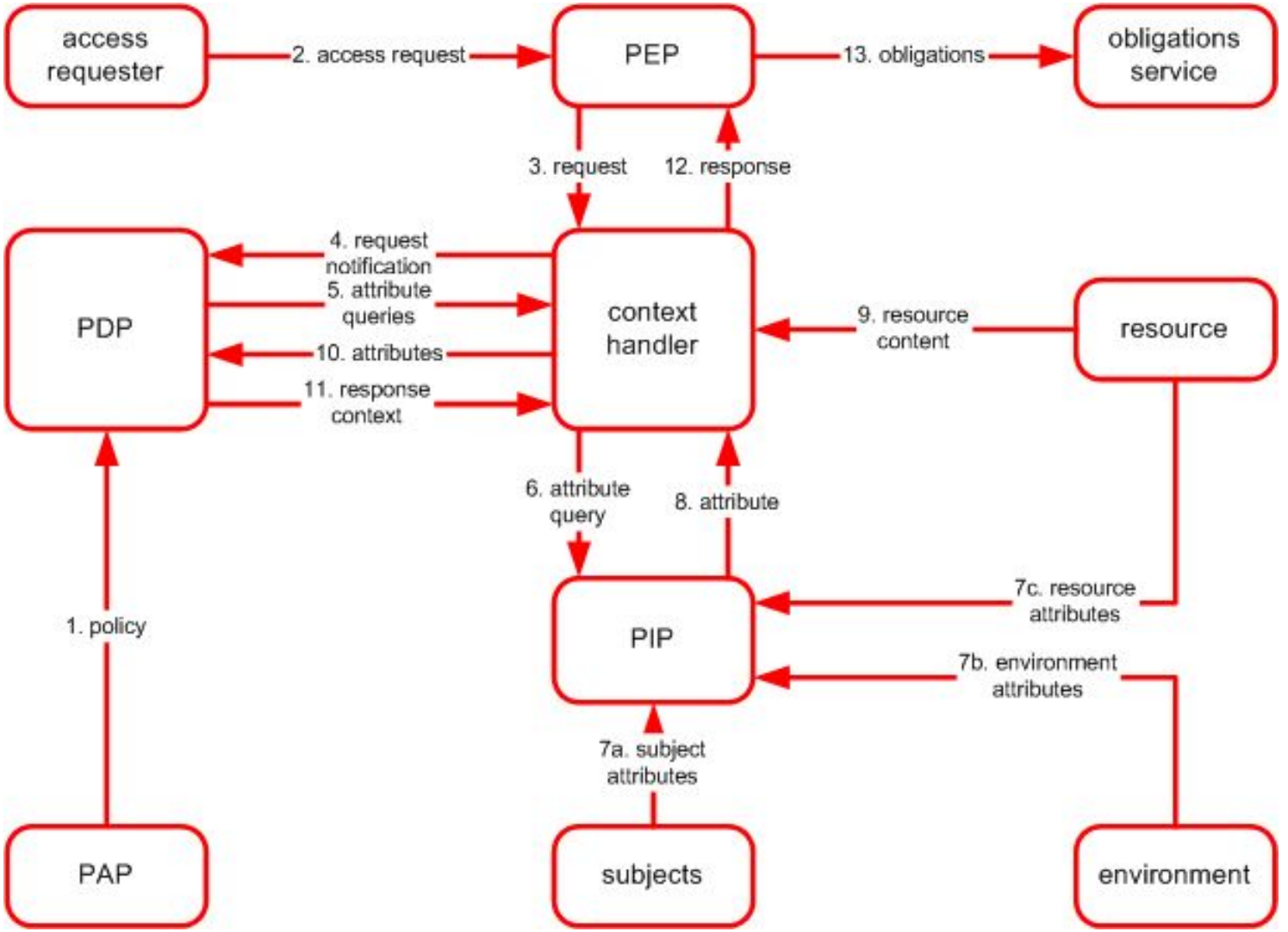
- Легко управлять и контролировать
- Раздельное назначение роли-пользователи и роли-привилегии
- Масштабируемость и иерархия
- Поддерживает принцип минимально необходимых привилегий
- Наследование и агрегирование привилегий/прав
  - Новая роль может включать комбинацию уже существующих ролей с их правами
- Упрощает процедуру делегирования

Инфраструктура управления привилегиями

PMI – Privilege Management Infrastructure (ISO/IEC 10181-3)

- Строится на основе Сертификатов Атрибутов (AC – Attribute Certificate)
- AC совместно с СОК определены стандартом X.509 version 4
  - СОК используется для аутентификации, AC используется для авторизации
- PMI как основа для построения RBAC
  - AC позволяет связать идентификатор пользователя с ролями и роли с привилегиями
  - Поддерживает иерархические системы RBAC, предоставляя возможность обьединения роли и дополнительных привилегий
  - Ограничивает глубину делегирования
- Политика PMI
  - Используется для контроля доступа к ресурсам на основе ролей
  - Правила определения ролей для пользователей и привилегий для ролей
  - Раздельные политики для субьекта, иерархия ролей, делегирование, др.

Основные компонены и потоки информации в PMI



PEP (Policy Enforcement Point)/
AEF (authorisation enforcement function)

PDP (Policy Decision Point)/ADF (authorisation decision function)

PIP (Policy Information Point)/AA (Attribute Authority)

PA – Policy Authority

Свободно распространяемые средства для AuthN/AuthZ

Разработаны в рамках проектов Internet2, FP5 и национальных научных сетей

- A-Select - http://a-select.surfnet.nl/
- Shibboleth - http://shibboleth.internet2.edu/
- PAPI - http://www.rediris.es/app/papi/index.en.html
- PERMIS (PrivilEge and Role Management Infrastructure Standards validation) - http://www.permis.org/
- SPOCP - http://www.spocp.org/

Для GRID-приложений

- VOMS – Virtual Organisation Management System
- GAAA Toolkit – http://www.aaaarch.org/

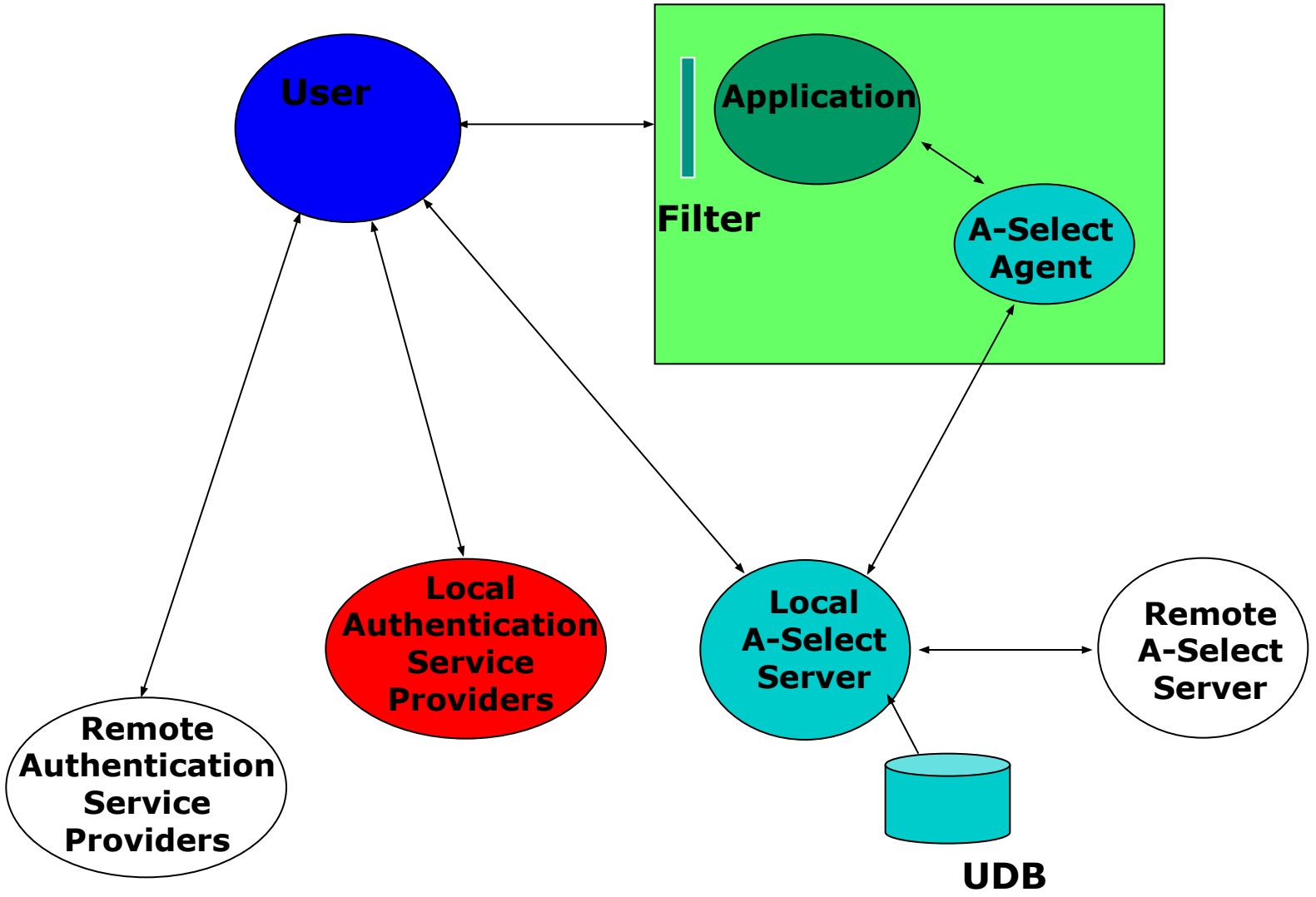
A-Select (1)

A-Select представляет собой распределенную систему веб-доступа (weblogin) с использованием cookie

Поддерживаемые методы аутентификации

- IP address
- User/password через RADIUS
- Банковская карточка (с режимом Internet banking – SMS/TAN, Challenge generator)
- SMS (mobile phone)
- LDAP
- PKI (в перспективе)

Компоненты A-Select



A-Select (2)

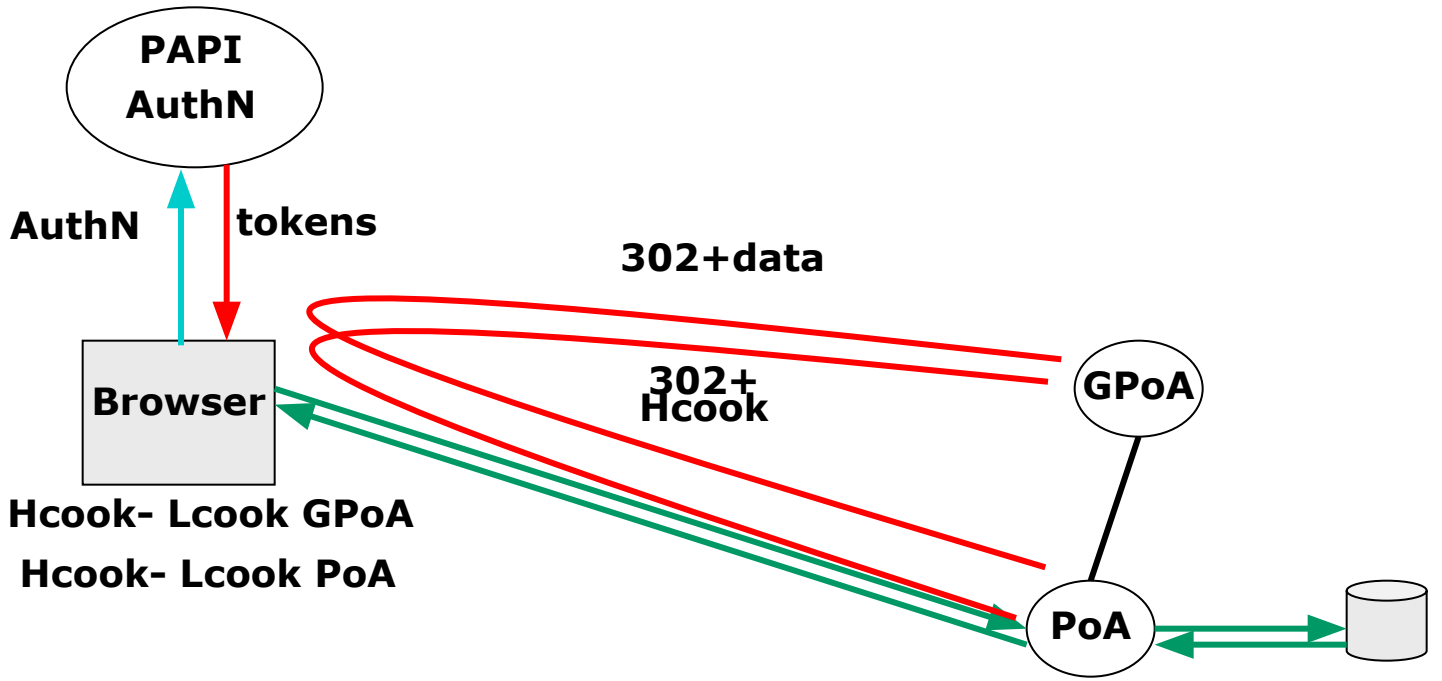
A-Select использует билеты/квитанции, которые содержат пользовательские мандаты/удостоверения.

- Есть два типа квитанций:
  - Квитанция, гарантирующая квитанцию ("ticket granting ticket"), выдаваемая после успешной аутентифиации ASP, and
  - Квитанция придложения ("application ticket"), которая выдается приложением, использующим A-Select.
- Единый доступ (Single-Sign-on) обеспечивается за счет назначения более длительного периода жизни для квитанция, гарантирующая квитанцию
- Квитанции A-Select реализованы как не-постоянные (non-persistent) cookie, которые сохраняются в браузере пользователя и видимы только для целевого сервиса или сервера

PAPI – простейшая система котроля веб-доступа

PAPI – распределенная система доступа к ресурсам Интернет

- Может использоваться как для доступа внутри реалма так между реалмами
- Использует HTTP/cookie и PKI



PERMIS (PrivilEge and Role Management Infrastructure Standards)

PERMIS обеспечивает авторизацию на основе политики с использованием Сертификатов атрибутов X.509 для хранения атрибутов/ролей пользователя

- Может работать с любой системой AuthN (например, username/pswd, PKI, Kerberos, etc.), которая обеспечивает AuthN и возвращает атрибуты пользователя
- На основе идентификатора пользователя (Субьекта), целевой системы (Цели), Действия возвращает решение о доступе на основе принятой Политики безопасности для данной цели
  - Политика и атрибуты пользователя могут быть запрошены сответственно из БД политики и хранилища атрибутов
- Политика основана на RBAC и описывается в форме XML подобной XACML
- PERMIS может работать в режиме push или pull (т.е., атрибуты передаются в PERMIS, или PERMIS запрашивает атрибуты от хранилища атрибутов)

SPOCP (Simple Policy Control Protocol)

Использует S-выражения для построения иерархической системы правил доступа для ролей

- Позволяет обрабатывать множество ролей и действий

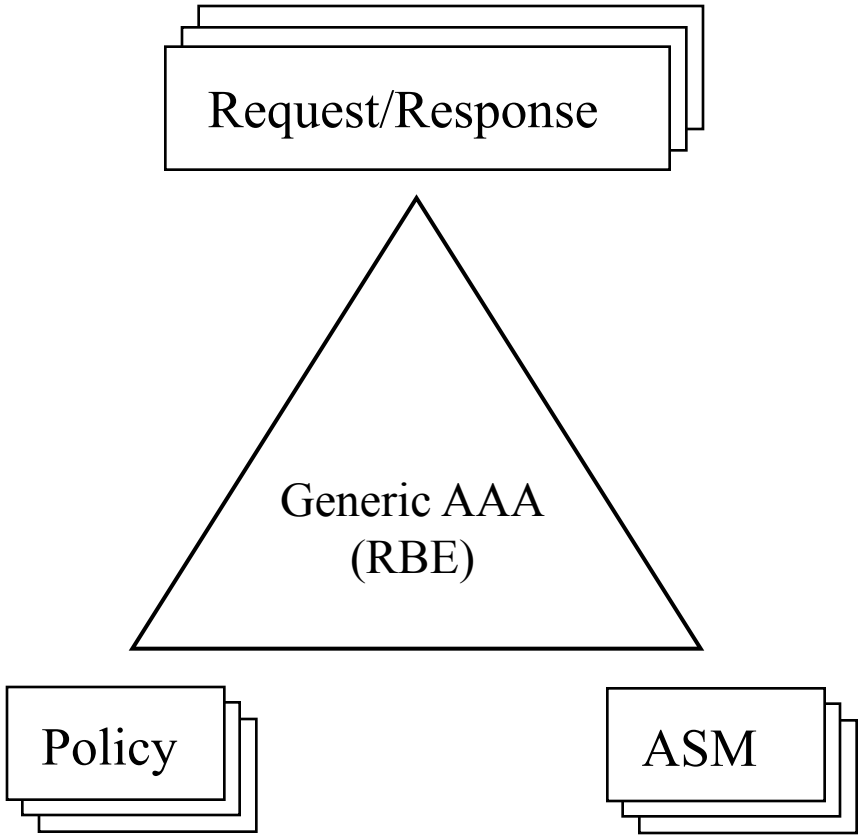
Пример:

Политика: (role UmU admin finance) <= (role UmU admin)

Запрос SPOCP от почтового SMTP сервера:
(spocp (resource mailrelay)(action mail)(subject (smtpauth roland)))

- SPOCP обеспечивает упрощенную процедуру обмена данными и запроса, а также упрощенный формат сообщений по сравнению со средствами общего назначения XACML и SAML

Базовая архитектура AAA



Policy based Authorization decision

- Req {AuthNtoken, ResourceCtx, Attr/Roles, PolicyTypeId}
- RBE (ReqCtx + Policy) =>
  => Decision {ResponseAAA, ActionExt}
- ActionExt = {ReqAAAExt, ASMcontrol}
- ResponseAAA = {AckAAA/RejectAAA, ReqAttr, ReqAuthN, BindAAA (Resource, Id/Attr)}

•Defined by Resource owner

•Translate logDecision => Action
•Translate State => LogCondition

GAAA Toolkit (1) – формат политики

```
if
( AuthN == TrustDomain )
then (
  if
  ( ResourceState == RequestResourceContext)
  then (
     if ((Action == action1))
     then (
       if ((Role == role1))
       then (
         Reply::Answer.Message = "Permit" )
       else ( Reply::Answer.Message = "Deny, Role is not valid" ) )
     else ( Reply::Answer.Message = "Deny, Action is not valid" ) )
  else ( Reply::Answer.Message = "Deny, Resource is not ready" )
else ( Reply::Answer.Message  = "Deny, Subject is not authenticated" )
```

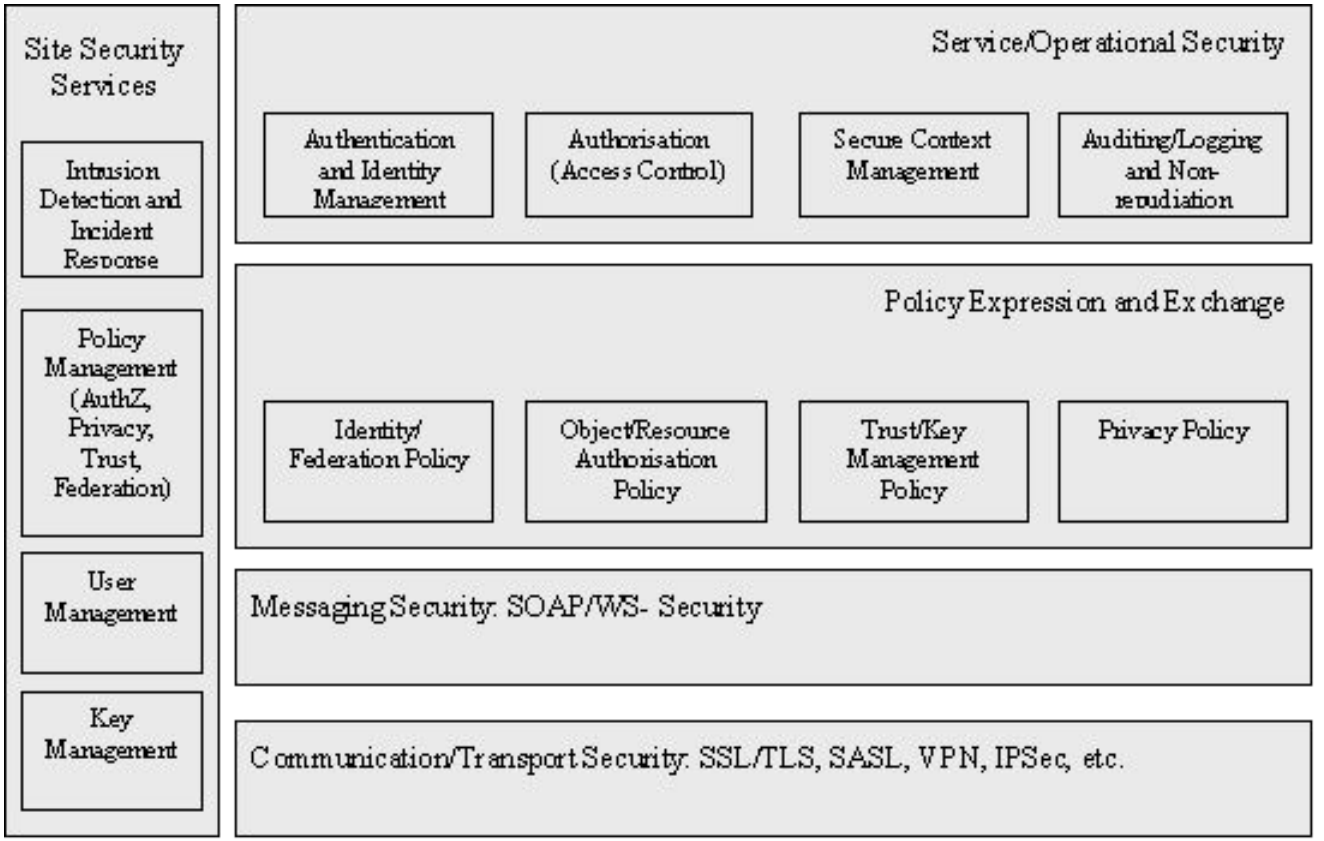
GAAA Toolkit (2)

- RBE реализована на J2EE (платформа Tomcat 4.5/5)
- Формат политики
  - Внутренний – GAAA
  - Обмен - XACML
- Автоматическая генерация политики на основе таблицы доступа
- Формат сообщений Запрос/Ответ - XACML

Примеры проектов - Collaboratory.nl (CNL)

- Консорциум: DSM, Corus, Philips, FEI, TI, UvA
- Создание безопасной среды для кооперативного выполнения сложных аналитических задач/экспериментов
  - Широкое использование XML Security, Web Services, Grid-технологий
- Архитектура безопасности на основе семантического описания работы/задачи
- Аутентификация – распределенная система управления идентификацией пользователей (Identity Management) на основе виртуальных организаций (BO)
  - Использование A-Select
- Управление доступом на основе ролей и политики доступа
  - Архитектура AAA и язык для управления доступом XACML

Аритектура безопасности CNL



Уровни услуг безопасности

1 - Сетевой/траспортный
включает сервисы безопасности сетевого уровня

2 – Безопасность сообщений
обеспечивает безопасность передачи XML-сообщений

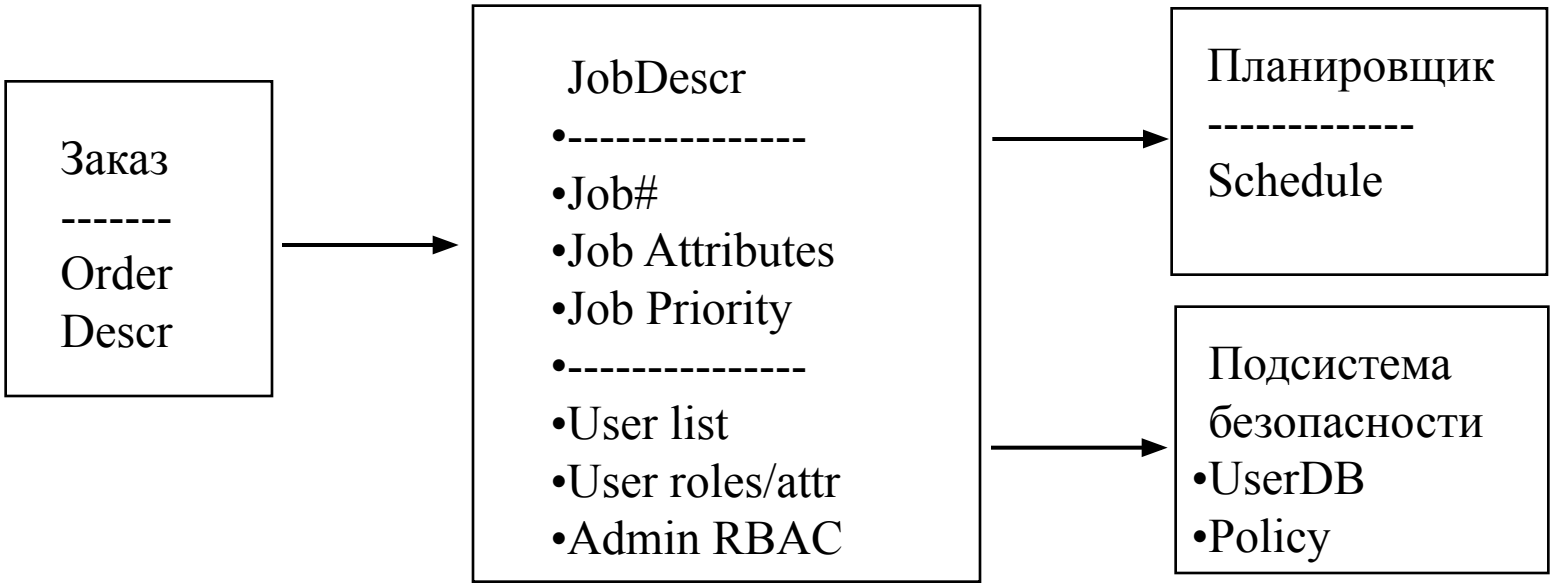
3 – Политика безопасности
обеспечивает применение политики безопасности ресурсов или ассоциаций для конкретных услуг безопасности

4 – Операционные сервисы безопасности
включает базовые/конечные сервисы безопасности, включая AuthN, AuthZ, ау дит

Услуги безопасности узла

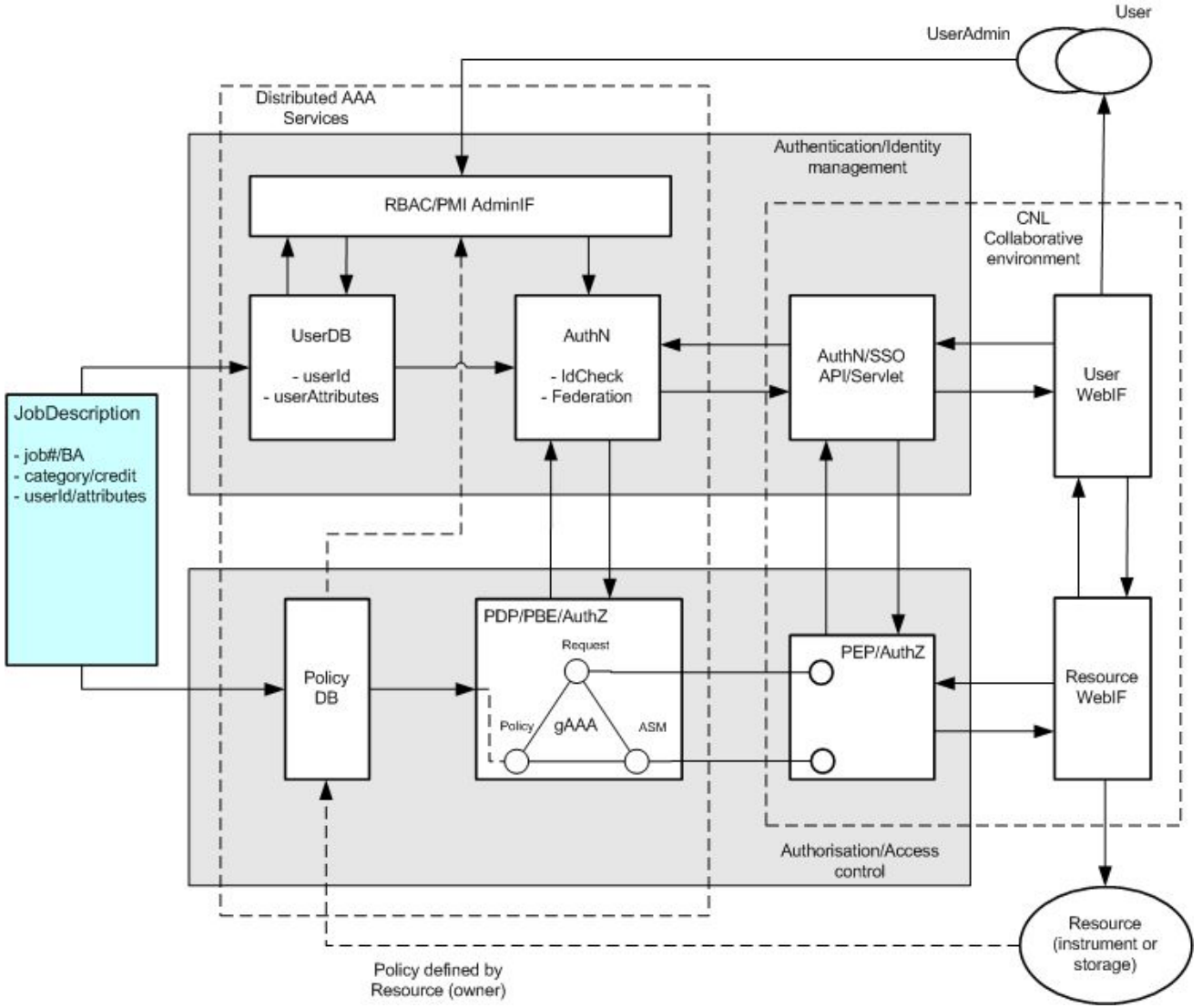
Управление ключами PKI, Управление пользователями, Управление политикой безопасности, Обнаружение вторжений в сеть и Реагирование на инциденты безопасности

CNL: Посстроене безопасности на основе описания работы



- Описание работы JobDescr является семантическим обьектом, определяющим атрибуты Работы, Пользователя и Политику безопасности
  - Требует технологии безопасности XML, построенные вокруг семантического обьекта
- Доверительный домен определяется бизнес-соглашением (BA - Business Agreement) или соглашением о доверии (TA - Trust Agreement) посредством COK/PKI

CNL: Процессы AuthN/AuthZ



PEP/AuthZ – принимает запрос от Ресурса или Пользователя и в стандартной форме передает его PDP/PBE

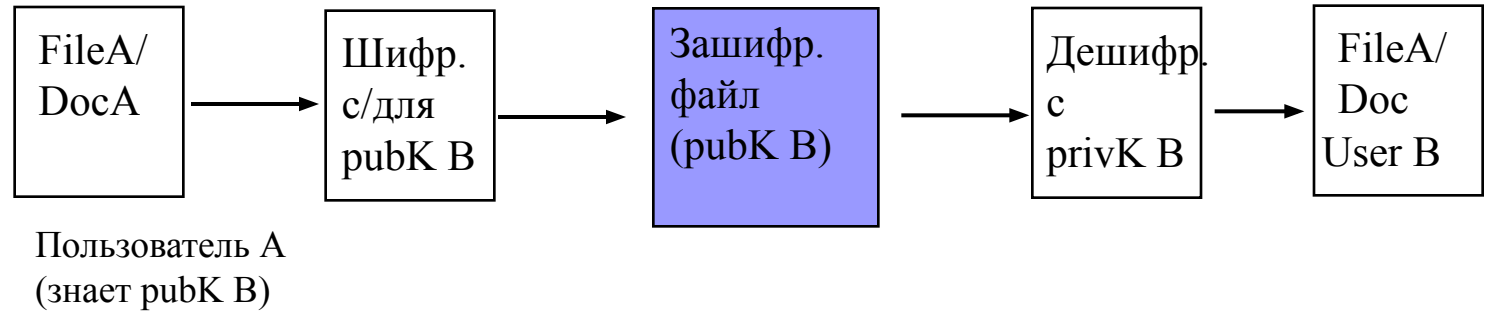
PDP/PBE – принимает решение о доступе на основе запроса (Субьект, Ресурс, Действие) и Политики

UserDB – хранит информацию о пользователе для AuthN (пароль, СОК) и атрибуты/роли пользователя для AuthZ

PolicyDB – БД политики безопасности для различных ресурсов

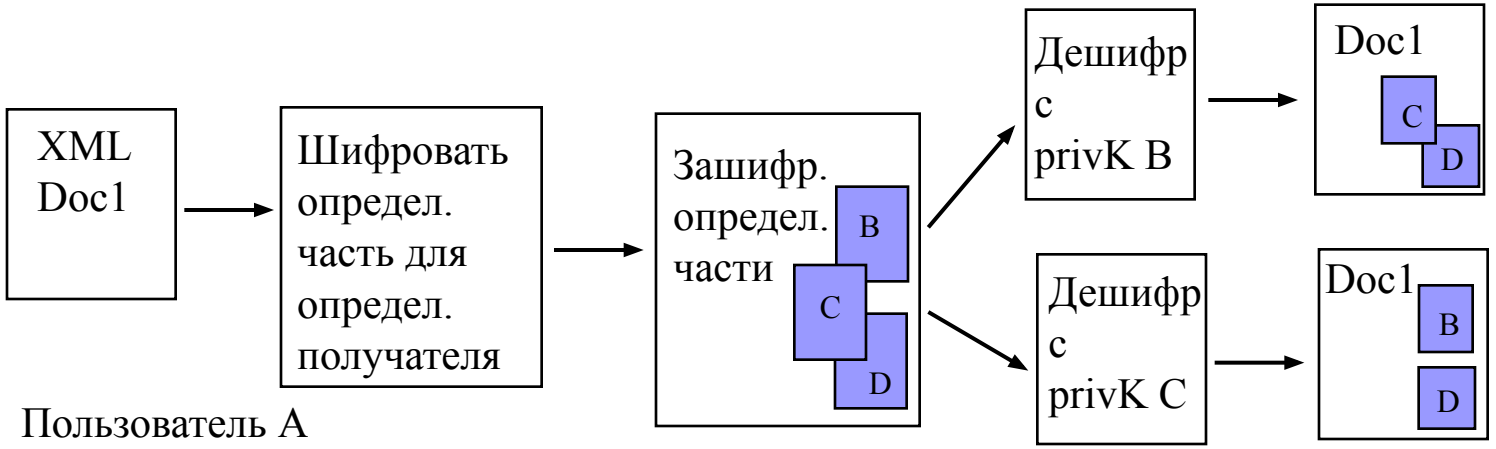
RBAC/PMI AdminIF – административный интерфейс для управления доступом пользователей

Шифрование файла и XML-шифрование



Только пользователь B может прочитать FileA при помощи privK B

Пользователь A
(знает pubK B)



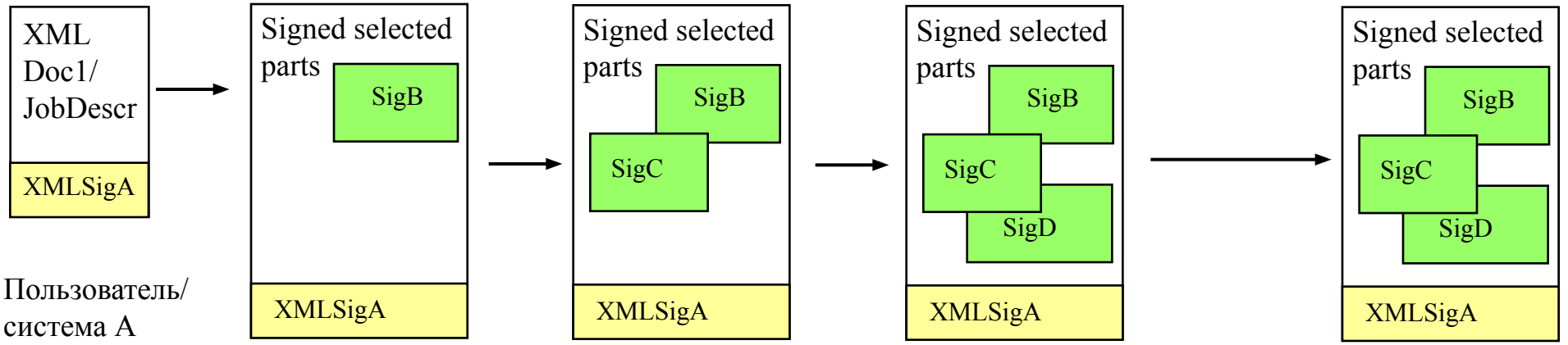
Пользователь B может проочитать весь Doc1 и дешифровать только часть B

Пользователь C может проочитать весь Doc1 и дешифровать только часть C

Пользователь A
(знает pubK B,C, D)

Для много-пользовательского шифрования Document может содержать ключ для дешифрации (симметричный или асимметричный), зашифрованный при помощи OK всех целевых получателей

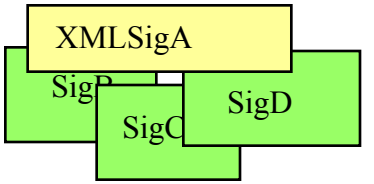
Связывание атрибутов с документом при помощи XMLSig



Пользователь/ система A создают XML Doc1 и подписывают его с SigA

Пользователи B, C, D подписывают определенные части документа своими секретными ключами privK B, C, D

- Новая ипформация может быть добавлена и документ подписан в целом
- Подпись может также включать другие подписи



Получатель проверяет целостность XML Doc1 посредством контроля цифровых подписей

XML Signature позволяет подписывать отдельные части документа

- Основа для аутетичности и целостности (Integrity and Authenticity)
- Связывание атрибутов безопасности и прав с документом или его частями

Примеры проектов –EGEE

JRA3 – Security Architecture and Services

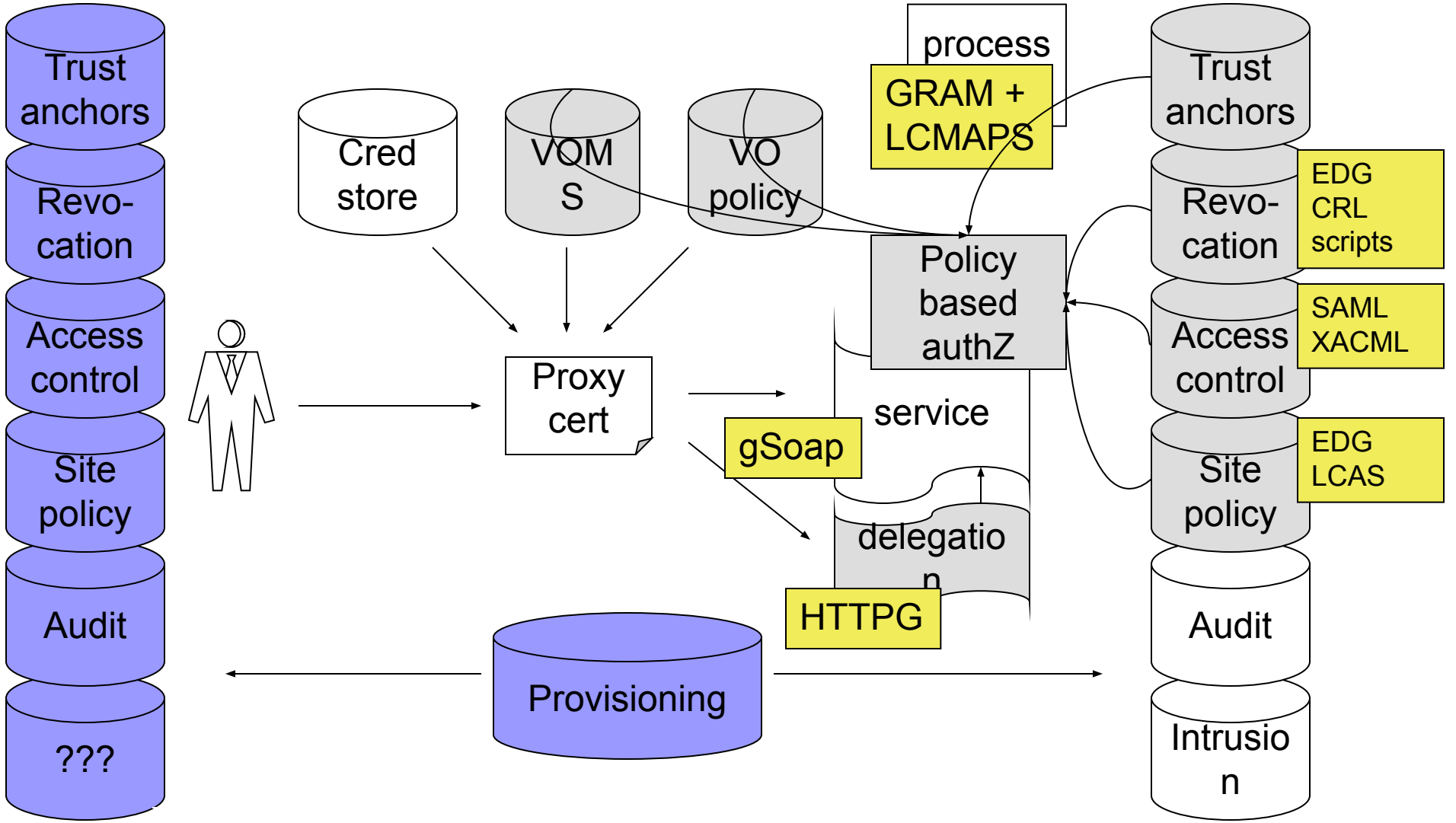
Статус:

- Определение требований и начало создания прототипа

Предложение:

- Потоколы транспортного уровня и уровня сообщений
  - SOAP over HTTPS или SOAP-XML Security over HTTP
- Делегирование
  - WSDL PortType
- Политика безопасности
  - Связывание с Грид-сервисами на уровне описания WSDL
  - Комбинирование политик ресурсов и виртуальных организаций
  - Интерфейс администратора политики

Примеры проектов – Безопасность в EGEE



Справочная информация

- Компоненты XML Security
- XML Web Services
- WS-Security
- OGSA Security

Компоненты безопасности XML - приложений

- XML Signature
- XML Encryption
- Декларации безопасности (Security Assertions)
  - SAML (Security Assertion Mark-up Language)
  - XrML (XML Right Mark-up Language)
  - XACML (XML Access Control Mark-up Language)
- XKMS (XML Key Management Specification)
- Архитектурные расширения
  - Web Services Security (WS-Security)
  - OGSA Security

Основные черты XML-подписи

Фундаментальная черта: возможность подписывать отдельные части документа так же как и целый документ.

- XML-документ может иметь длинную историю, при этом различные части докумепнта могут создаваться и визироваться различными субьектами и в различное время
- Различные стороны/субьекты могут иметь полномочия подписывать только различные части документа
- Позволяет сохранять целостность одних частей документа и иметь возможность изменять другие части документа
- Позволяют присоединять маркеры/мандаты безопасности к документу в отличие от использования безопасного соединения клиент/сервер
- XML-подпись обеспечивает сервисы безопасности для протколов, основанных на XML
  - А также основу для включения информации о состоянии

Структура XML-подписи

```
<Signature ID?>
   <SignedInfo>
    <CanonicalizationMethod/>
    <SignatureMethod/>
    (<Reference URI? >
        (<Transforms>)?
        <DigestMethod>
        <DigestValue>
    </Reference>)+
   </SignedInfo>
   <SignatureValue>
   (<KeyInfo>)?
   (<Object ID?>)*
</Signature>
```

Расширение архитектуры XML-безопасности

WS-Security (Web Services Security)

- Расширения к формату сообщений SOAP (Simple Object Access Protocol)
  - Стандартные заголовки для аутентификации и авторизации, аудита, ассоциаций безопасности, приватности
    - Обмена удостоверяющими мандатами/маркерами в формате X.509 PKC, SAML, XrML, XCBF
  - Цифровая подпись, шифрование
- Протоколы для синхронного и асинхронного обмена сообщениями и организации междоменных кооперативных Веб-сервисов

OGSA Security (Open Grid Services Architecture)

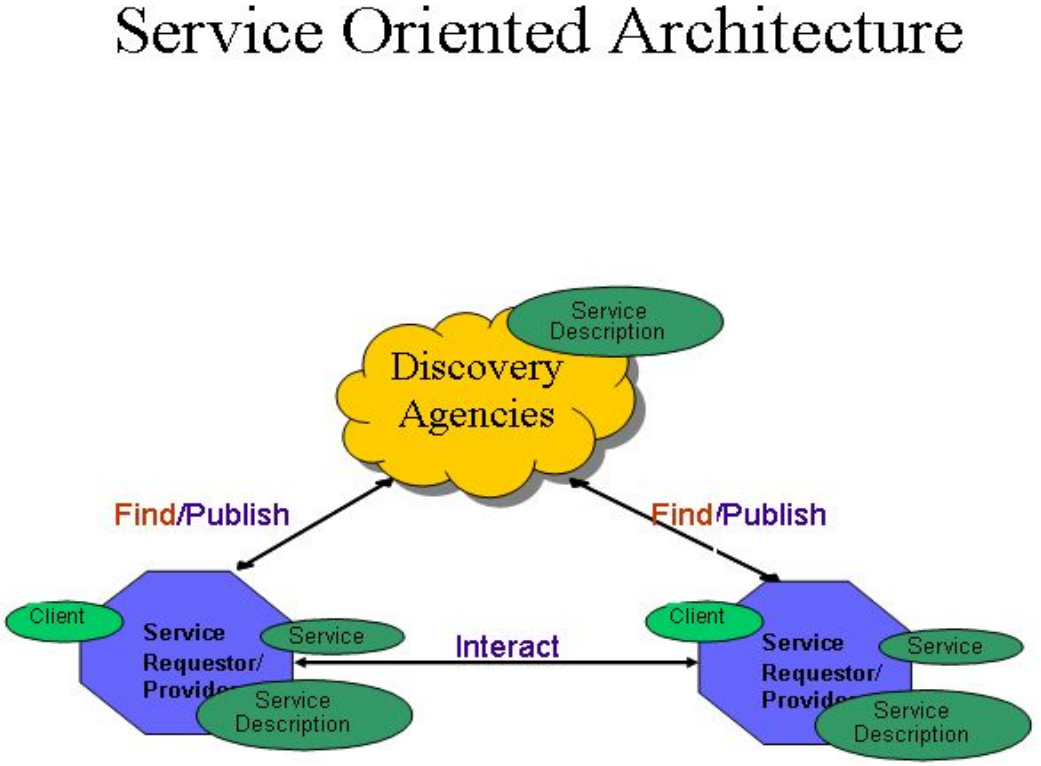
- Построена на основе WS-Security
- Функциональность для создания виртуальных организаций (ВО)
  - Делегирования полномочий (credentials) и федерация идентификаторов субъекта
  - Анонимность/приватность, ассоциации для управления доступом
- Поддержка транзитивных процессов с конечными состояниями (transitional stateful processes)

Liberty Alliance и сетевая идентификация

Liberty Alliance Project (LAP)

- LAP вводит понятия провайдера идентификации (identity provider) и круга доверия (trust circle)
- LAP обеспечивает полный контроль за своим идентификатором и ассоциациями/федерациями, которые могут создаваться провайдерами идентификации – на основании предварительного согласия пользователя
- LAP использует SAML и расширяет его новыми элементами и протоколами
- LAP определяет три модели доверия, основанные на PKI или бизнес-отношениях: взаимное, через посредника-брокера и среди сообщества
- Основные функции LAP: Федерация идентификации; Представление провайдера идентификации; аутентификация; использование псевдонимов и поддержка анонимности; глобальный выход из системы

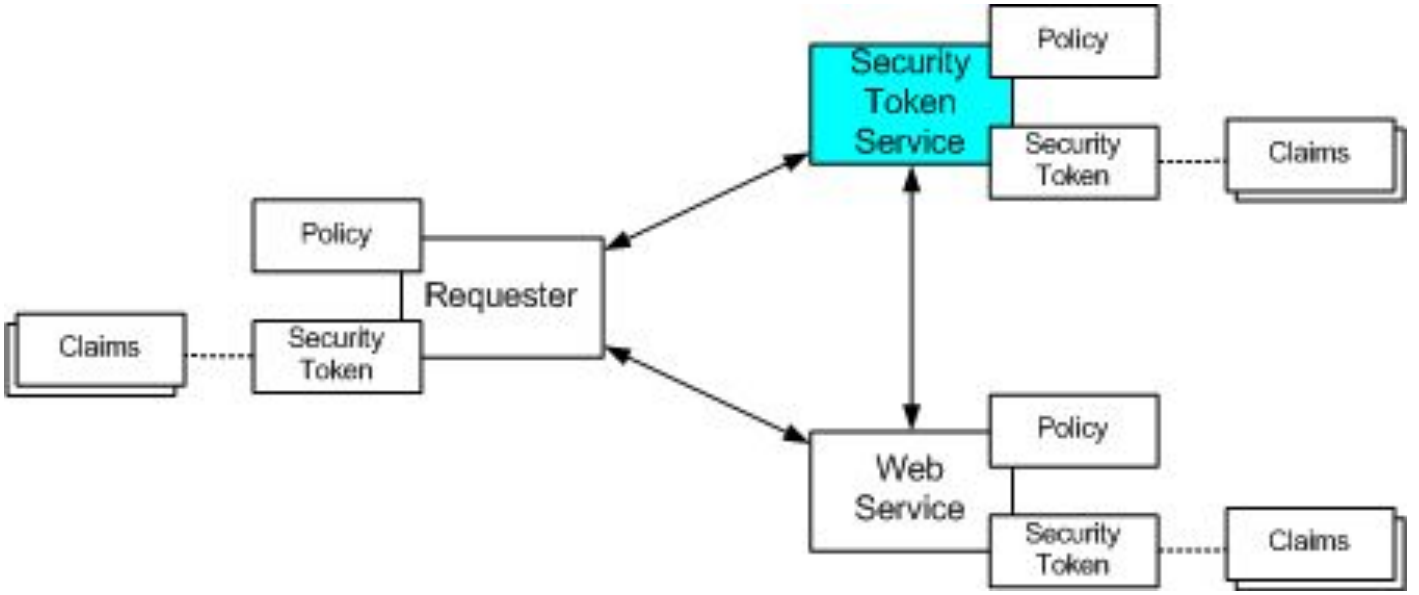
Архитектура на основе XML Web Service



- Описание на основе WSDL (Web Services Description Language)
- Обмен сообщениями в формате SOAP при помощи протоколов HTTP, SMTP, TCP, etc.
- Публикация и поиск посредством UDDI

Веб-сервис – программная система, идентифицируемая URI, интерфейс внешнего доступа которой and bindings описываются при помощи XML. Другие программные системы могут обнаруживать и взаимодействовать с Веб-сервисами в соответствии с их описанием на основе использования XML-сообщений посредством протоколов Интернет.

Модель безопасности Web Services



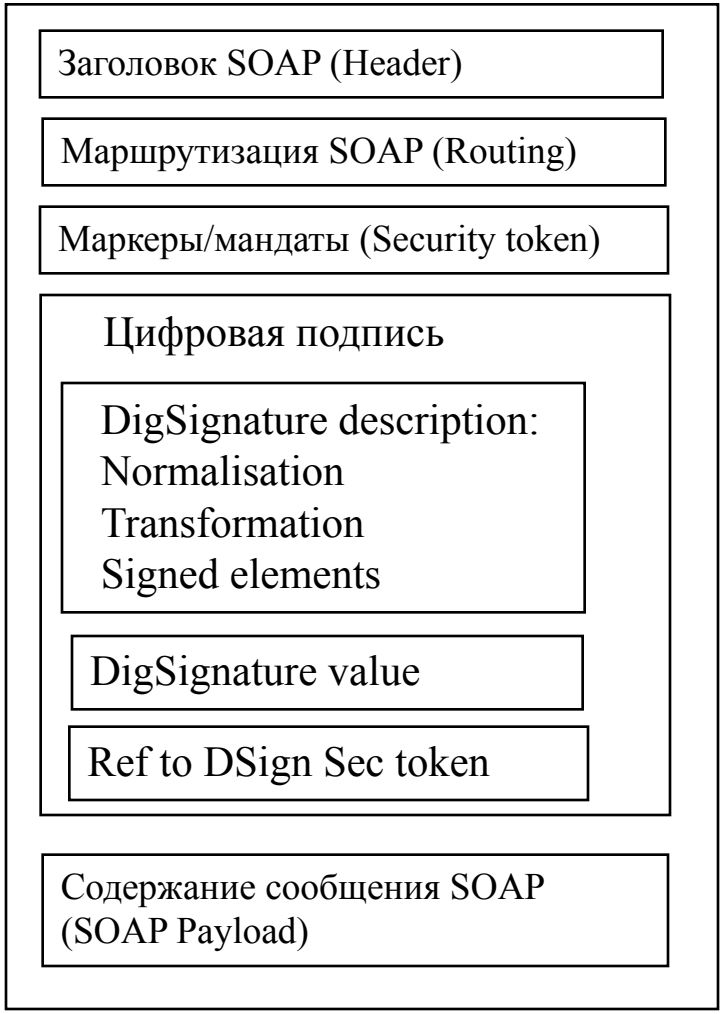
Security token types

- Username/password
- X.509 PKC
- SAML
- XrML
- XCBF

WS-Security: describes how to attach signature and encryption headers to SOAP messages. In addition, it describes how to attach security tokens, including binary security tokens such as X.509 certificates, SAML, Kerberos tickets and others, to messages.

Core Specification - Web Services Security: SOAP Message Security
http://www.oasis-open.org/committees/download.php/1043/WSS-SOAPMessageSecurity-11-0303.pdf

WS-Security: Расширения к формату SOAP



URI: http://schemas.xmlsoap.org/ws/2002/04/secext

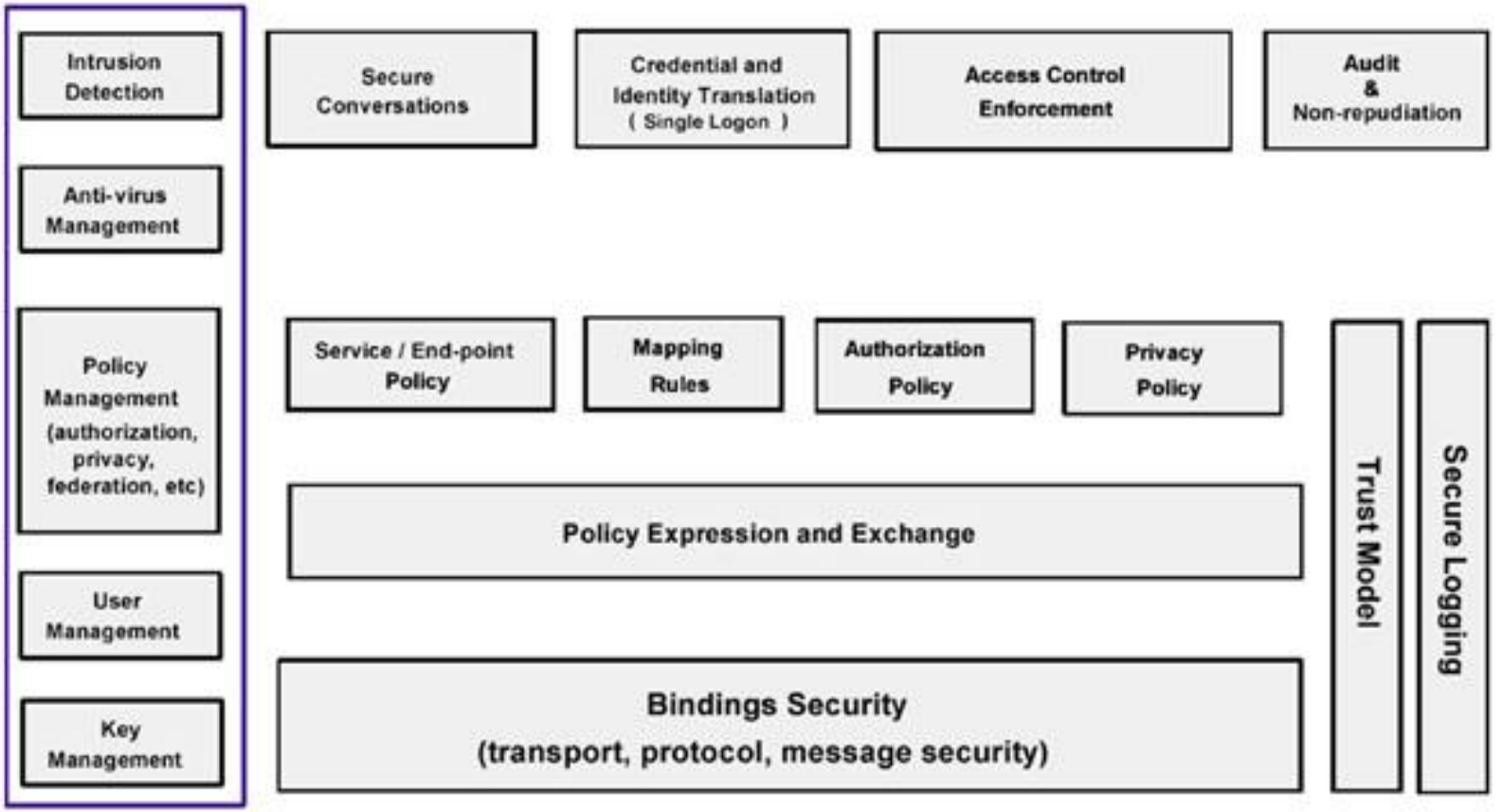
Пространства имен, используемые в WS-Security:

SOAP	S	http://www.w3.org/2001/12/soap-envelope
XML Digital Sign	ds	http://www.w3.org/2000/09/xmldsig#
XML Encryption	xenc	http://www.w3.org/2001/04/xmlenc#
XML/SOAP Routing	m	http://schemas.xmlsoap.org/rp
WSSL	wsse	http://schemas.xmlsoap.org/ws/2002/04/secext

Элементы безопасности

- Заголовок определяет конечного получателя/исполнителя
- Допускаются множественные заголовки/получатели
- Новые заголовки

Архитектура безопасности OGSA Security



Security Component Layering

Построена на основе WS-Security

Proxy Certificate Profile

- Impersonation – used for Single-Sign-On and Delegation
  - Unrestricted Impersonation
  - Restricted Impersonation defined by policy
- Proxy with Unique Name
  - Allows using in conjunction with Attribute Cert
  - Used when proxy identity is referenced to 3rd party, or interact with VO policy
- Limited validity time – approx. 24 hours

Proxy Certificate (PC) properties:

- It is signed by either an X.509 End Entity Certificate (EEC), or by another PC. This EEC or PC is referred to as the Proxy Issuer (PI).
- It can sign only another PC. It cannot sign an EEC.
- It has its own public and private key pair, distinct from any other EEC or PC.
- It has an identity derived from the identity of the EEC that signed the PC.
- Although its identity is derived from the EEC's identity, it is also unique.
- It contains a new X.509 extension to identify it as a PC and to place policies on the use of the PC. This new extension, along with other X.509 fields and extensions, are used to enable proper path validation and use of the PC.

Reference: PKI Basics

PKI (Public Key Infrastructure) – Инфраструктура открытых ключей (ИОК)

- Связывает идентификатор (имя собственное, distinguished name) субъекта с его открытым ключом
- Основа ИОК – Сертификат открытого ключа (СОК, PKC - Public Key Certificate)
  - CRL – Certificate Revocation List
- Компоненты ИОК
  - Identification Service (IS)
  - Registration Authority (RA)
  - Certification Authority (CA)
  - Certificate Repository (CR), normally built on LDAP

Reference: PKC vs AC: Purposes

- X.509 PKC binds an identity and a public key
- AC is a component of X.509 Role-based PMI
  - AC contains no public key
  - AC may contain attributes that specify group membership, role, security clearance, or other authorisation information associated with the AC holder
  - Analogy: PKC is like passport, and AC is like entry visa
- PKC is used for Authentication and AC is used for Authorisation
  - AC may be included into Authentication message
- PKC relies on Certification Authority and AC requires Attribute Authority (AA)

PKC vs AC: Certificates structure

X.509 PKC

- Version
- Serial number
- Signature
- Issuer
- Validity
- Subject
- Subject Public key info
- Issuer unique identifier
- Extensions

AC

- Version
- Holder
- Issuer
- Signature
- Serial number
- Validity
- Attributes
- Issuer unique ID
- Extensions

X.509 PKC Fields and Extensions – RFC 3280

X.509 PKC Fields

- Serial Number
- Subject
- Subject Public Key
- Issuer Unique ID
- Subject Unique ID

X.509 PKC Fields

- Private Extensions
  - Authority Information Access
  - Subject Information Access
- Custom Extensions

X.509 PKC Extensions

- Standard Extensions
  - Authority Key Identifier
  - Subject Key Identifier
  - Key Usage
  - Extended Key Usage
  - CRL Distribution List
  - Private Key Usage Period
  - Certificate Policies
  - Policy Mappings
  - Subject Alternative Name
  - Issuer Alternative Name
  - Subject Directory Attributes
  - Basic Constraints
  - Name Constraints

AC Attribute Types and AC Extensions

AC Attribute Types

- Service Authentication Information
- Access Identity
- Charging Identity
- Group
- Role
- Clearance
- Profile of AC

AC Extensions

- Audit Identity
- To protect privacy and provide anonymity
- May be traceable via AC issuer
- AC Targeting
- Authority Key Identifier
- Authority Information Access
- CRL Distribution Points