




**JBoss Security**

Anil Saldhana  
 asaldhan@redhat.com




### Speaker Introduction

- Anil Saldhana is a Senior Software Engineer in the Core R & D group of JBoss at Red Hat.
  - Currently focused on JBoss Security and Identity Management for JEMS.
  - Previously worked in Tomcat and Web Services areas.
  - JSR-196 Expert Group member.



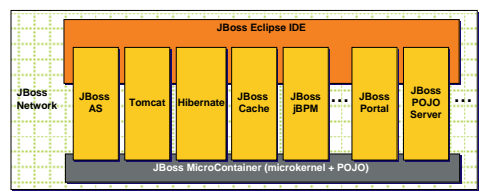

### Agenda

- JEMS Architecture
- Service Provider Interface (SPI)
- Authentication Infrastructure
  - JSR 196 (Java Authentication SPI for Containers)
  - JBoss Federated SSO
- Authorization Infrastructure
  - JACC, XACML and use cases
  - Authorization Framework in JBoss 5.0
- Mapping Framework
- Auditing Service
- Roadmap and Future Possibilities
- Demo/Q &A
- Resources




### JEMS Architecture

- Security is a cross-cutting concern


### Service Provider Interface (SPI)

- Security Project broken into
  - Service Provider Interface (SPI)
  - Implementation of the SPI
    - JBossSX (Implementation of the SPI for the JBoss Application Server)
    - Your own implementation? Why not?
- SPI includes
  - Authentication Manager
  - Authorization Framework
  - Mapping Framework
  - Auditing Framework



### Authentication Infrastructure

- JAAS Based Framework
- JSR-196 (Java Authentication SPI for Containers)
  - Allows us to have a notion of container messages during authentication, unlike the current JAAS based infrastructure in JBoss.
  - Web(HttpServletRequest/HttpServletResponse) & Soap(SOAPMessage)
  - Targeted for JEE6
- JBoss Federated SSO Project
  - v1.0.0 Beta version released
  - Framework for Federation with Tomcat and JBoss Security integration



## Authentication Infrastructure

- Authentication Manager Interface

```
public interface AuthenticationManager
{
    public boolean isValid(Principal principal, Object credential);

    /**
     * Trust related usecases may require translation of a principal from another domain
     * to the current domain
     * An implementation of this interface may need to do a backdoor contact of the external
     * trust provider in deriving the target principal
     * @param anotherDomainPrincipal Principal that is applicable in the other domain
     * (Can be null - in which case the contextMap is used
     * solely to derive the target principal)
     * @param contextMap Any context information (including information on the other domain
     * that may be relevant in deriving the target principal). Any SAML
     * assertions that may be relevant can be passed here.
     * @return
     */
    Principal getTargetPrincipal(Principal anotherDomainPrincipal, Map contextMap);
}
```



## Authorization Infrastructure

- Difficult space to solve compared to authentication.
  - Authentication - Who are you?
    - usually userid/password, certs etc.
  - Authorization - What are you allowed to do?
    - servlet spec: based on resource uris.
    - Ejb spec: based on ejb methods.
    - JMS needs: maybe based on destinations.
    - custom needs: allow this part of the app if you are the CEO's son.
- Java Authorization
  - Security Policy files and Java Permissions.
  - JACC uses this paradigm in the JEE world.



## Authorization Infrastructure

- JACC [Java Authorization Contract for Containers]
  - JSR-115
  - Only specification mandated in the JEE world.
    - JEE 1.4 (JACC 1.0) and JEE 1.5 (JACC 1.1)
  - Defines Java Permissions for Servlet and EJB specifications.
  - Possible to extend this permission model for custom needs.
    - JBoss Portal v2.4 has a custom JACC permission model.
    - Extensions are not spec supported.

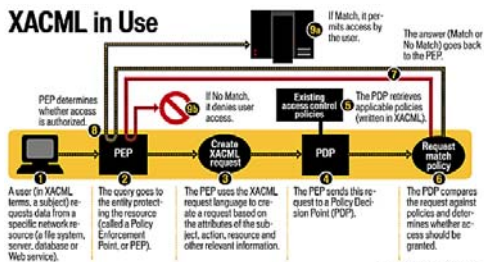


## Authorization Infrastructure

- XACML [eXtensible Access Control Markup Language]
  - OASIS Standard
  - Rich language for Access Control
    - Can use all available information for access decision - Resource properties, Environmental conditions (date/time/location) and Subject Attributes
  - Cons
    - Creating/editing xacml policy files is cumbersome with no good tools available



## Authorization Infrastructure



## Authorization Infrastructure

- XACML Use Case for JBoss Portal
  - Let us define the company policy as follows
    - /companyportal
      - portal accessible only between 9am and 5pm
    - /companyportal/EighteenYearOld
      - portal page accessible only if age of the subject == 18 years



## XACML Use Case – Policy

```
<?xml version="1.0" encoding="UTF-8"?>
<Policy xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" PolicyId="..."
  RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:deny-overrides">
  <Description>Policy for Portal Use Case.</Description>
  <Target>
  <Rule RuleId="urn:oasis:names:tc:xacml:2.0:test:tl:rule" Effect="Permit">
  <Description>Portal accessible between 9 am and 5pm.</Description>
  <Target>
  <Resources>
  <Resource>
  <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:anyURI-equal">
  <AttributeValue
  DataType="http://www.w3.org/2001/XMLSchema#anyURI">http://host1.companyportal/</AttributeValue>
  <ResourceAttributeDesignator
  AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
  DataType="http://www.w3.org/2001/XMLSchema#anyURI">
  </ResourceMatch>
  </Resource>
  </Resources>
  </Target>
  </Rule>
  </Policy>
```



## XACML Use Case – Policy

```
<Condition FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:time-greater-than-or-equal">
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:time-one-and-only">
  <EnvironmentAttributeDesignator DataType="http://www.w3.org/2001/XMLSchema#time"
  AttributeId="urn:oasis:names:tc:xacml:1.0:environment:current-time" />
  </Apply>
  <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#time">09:00:00</AttributeValue>
  </Apply>
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:time-less-than-or-equal">
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:time-one-and-only">
  <EnvironmentAttributeDesignator DataType="http://www.w3.org/2001/XMLSchema#time"
  AttributeId="urn:oasis:names:tc:xacml:1.0:environment:current-time" />
  </Apply>
  <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#time">17:00:00</AttributeValue>
  </Apply>
  </Apply>
  </Condition>
</Rule>
```



## XACML Use Case – Policy

```
<Rule RuleId="urn:oasis:names:tc:xacml:2.0:jboss-test:tl:rule" Effect="Permit">
<Description>The EighteenYearOld page accessible if you are 18.</Description>
<Target>
<Resources>
<Resource>
<ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:anyURI-equal">
<AttributeValue
DataType="http://www.w3.org/.../anyURI">http://host1.companyportal/EighteenYearOld/</AttributeValue>
<ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-
id" DataType="http://www.w3.org/.../anyURI">
</ResourceMatch>
</Resource>
</Resources>
</Target>
<Condition>
<Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:integer-equal">
<Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:integer-one-and-only">
<SubjectAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:jboss-test:age"
DataType="http://www.w3.org/2001/XMLSchema#integer">
</Apply>
<AttributeValue DataType="http://www.w3.org/2001/XMLSchema#integer">18</AttributeValue>
</Apply>
</Condition>
</Rule>
</Policy>
```



## XACML Use Case – Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os" ...>
<Subject>
<Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id" DataType="...#string">
<AttributeValue>Anil Salghana</AttributeValue>
</Attribute>
</Subject>
<Resource>
<Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id" DataType="http://www.w3.org/.../anyURI">
<AttributeValue>http://host1.companyportal/</AttributeValue>
</Attribute>
</Resource>
<Action>
<Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id" DataType="...#string">
<AttributeValue>read</AttributeValue>
</Attribute>
</Action>
<Environment>
<Attribute
AttributeId="urn:oasis:names:tc:xacml:1.0:environment:current-time" DataType="http://www.w3.org/.../time">
<AttributeValue>09:23:47.05.00</AttributeValue>
</Attribute>
</Environment>
</Request>
```



## XACML Use Case – Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response
  xmlns="urn:oasis:names:tc:xacml:2.0:context:schema:os"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:context:schema:os
  access_control-xacml-2.0-context-schema-os.xsd">
  <Result>
  <Decision>Permit</Decision>
  <Status>
  <StatusCode
  Value="urn:oasis:names:tc:xacml:1.0:status:ok"/>
  </Status>
  </Result>
</Response>
```



## Authorization Infrastructure

- Authorization Framework
  - Part of JBoss 5.0, starting Beta
  - Pluggable authorization provided via authorization modules.
    - Authorization Modules are similar to JAAS Login Modules.
    - Modules can implement JACC, XACML or any custom logic.
    - Module option can be REQUIRED, REQUISITE, SUFFICIENT or OPTIONAL.
  - JBossSX – configurable at the security domain level in conf/login-config.xml

```
public class AuthorizationContext
{
  /**
   * Authorize the Resource
   * @param resource
   * @return AuthorizationContext.PERMIT or AuthorizationContext.DENY
   * @throws AuthorizationException
   */
  public int authorize(final Resource resource) throws AuthorizationException
}
```



## Authorization Infrastructure

### • Authorization Manager Interface

```
public interface AuthorizationManager
{
    public int authorize(Resource resource);
}
/*
 * Trust uscases may have a need to determine the roles of the target
 * principal which has been derived via a principal from another domain
 * by the Authentication Manager
 * An implementation of this interface may have to contact a trust provider
 * for additional information about the principal
 * @param targetPrincipal Principal applicable in current domain
 * @param contextMap Read-Only Contextual Information that may be useful for the
 * implementation in determining the roles.
 * @return
 */
public Group getTargetRoles(Principal targetPrincipal, Map contextMap);
}
```



## Service Provider Interface (SPI)

### • Mapping Framework

- Role Mapping
  - Map a set of roles for the subject based on some criteria (current principals in the subject or deployment level or any other criteria)
- Principal Mapping
  - Identity Mapping – map a token to a principal
  - Certificate Mapping – map a X509 certificate to a principal
- JBossSX – configurable at the Security Domain level in conf/login-config.xml

```
public class MappingContext
{
}
/*
 * Apply mapping semantics on the passed object
 * @param obj Read-only Contextual Map
 * @param mappedObject an object on which mapping will be applied
 */
public void performMapping(Map obj, Object mappedObject);
}
```



## Service Provider Interface (SPI)

### • Auditing Framework

- Configurable at the security domain level in JBossAS
- Audit Providers that are plugged in can audit the event at various levels or any conditions
- First cut of the prototype is
  - a logging provider that has a category in conf/log4j.xml
  - audits at the TRACE level in logs/security/audit.log (location can be changed)

```
public class AuditContext
{
    public AuditContext(String securityDomainName)
    public void audit(AuditEvent ae)
}
}
```



## Service Provider Interface (SPI)

### • Final Thoughts

- remember in JBoss 3.2.x/4.0.x, the authenticated subject was populated with the roles in a Group Principal called as "Roles"
  - now, we establish a Security Context (SC)
  - JBoss Authorization Manager uses the mapping framework to establish the SC roles
  - JBoss Authorization Manager uses the Authorization Framework to make a decision
- If you are integrating into the JBoss Micro Container, then you should be able to plug-in your implementation of the Security SPI or at the least
  - provide modules for authorization, various mapping or auditing



## Roadmap and Future Possibilities

- Security v2.0.0.Beta has been released
  - Mainly for JBoss 5.0.0.Beta1
  - Contains the SPI and JBossSX implementation
- Security v2.0.0
  - Support for federation via the enhanced authentication and authorization manager interfaces
  - SPI/implementation cleanup based on feedback
- Security v2.1.0
  - Support for SAML 2.0, WS-Trust etc
  - JBoss Federated SSO will synergize with Security 2.1.0
    - Will continue separate implementation for JBoss 4.0.x



## Demo/Q&A



## Resources

- JBoss Security Project
  - <http://labs.jboss.com/portal/jbosssecurity>
- JBoss Federated SSO Project
  - <http://labs.jboss.com/portal/jbosssso>
- **Always looking for volunteers from the community!!!**

