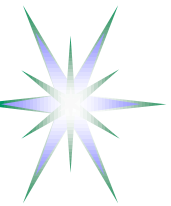
Security Services Lifecycle Management in On-Demand Infrastructure Services Provisioning

Yuri Demchenko
System and Network Engineering Group
University of Amsterdam

CPSRT 2010 Workshop @ CloudCom 2010
30 October - 3 December 2010, Indianapolis



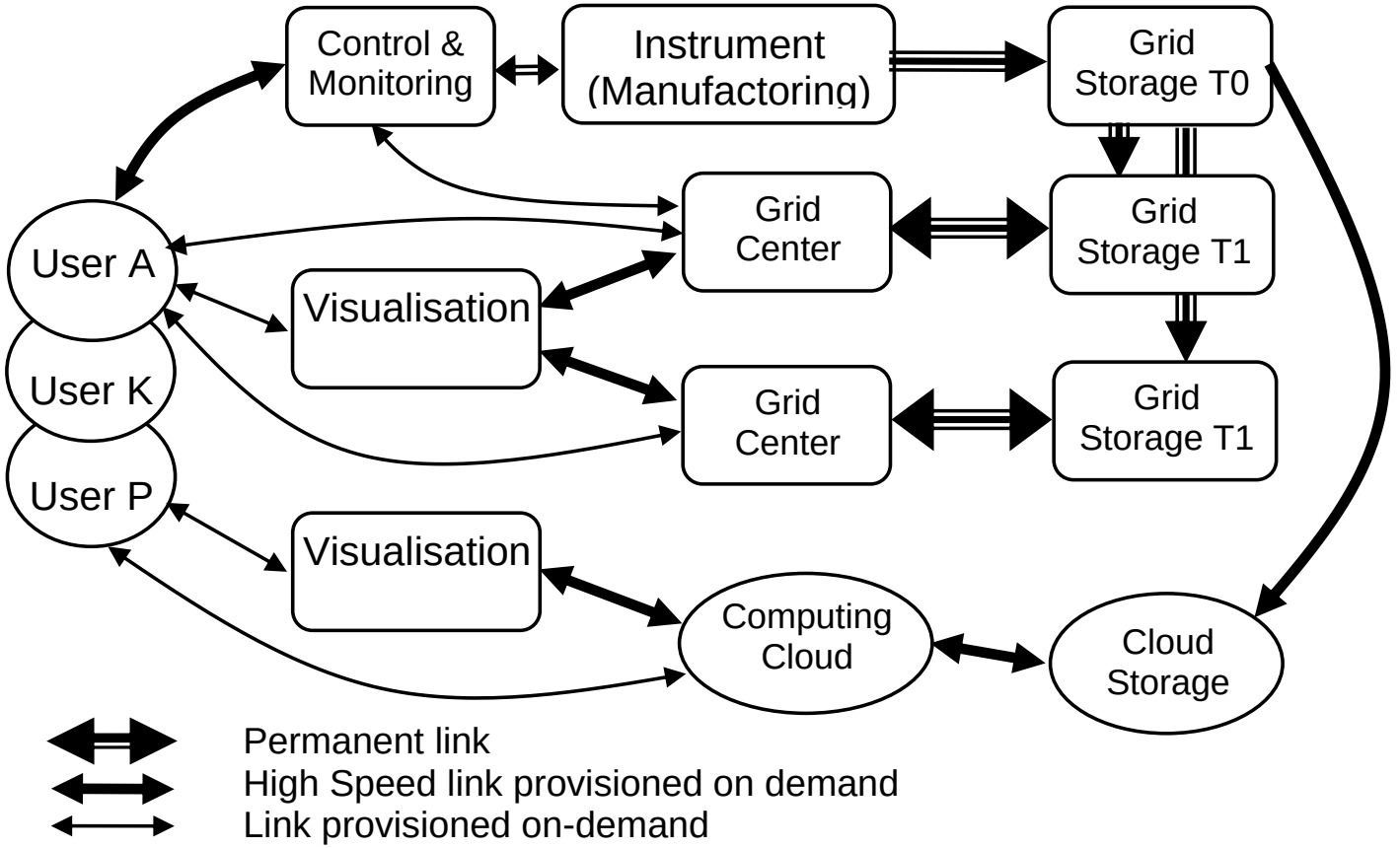
Outline

- Background for this research
- On-Demand Infrastructure Services Provisioning and Composable Services Architecture (CSA)
  - CSA Service Delivery Framework and Services Lifecycle Management
- Proposed Security Services Lifecycle Management and related security mechanisms
- Implementation – GAAA Toolkit and Security sessions management
- Summary and Discussion

Background to this research

- Current projects
  - GEANT3 JRA3 Task 3 Composable Services
    - European NREN infrastructure
  - GEYSERS – On-demand Optical + IT infrastructure resources provisioning
    - Wide participation from large European network (Telefonika, Alcatel-Lucent, Interoute) and application providers (SAP)
- Past projects
  - EGEE Grid Security middleware – gLite pluggable Java Authorisation Framework
  - Phosphorus project Security architecture for multi-domain Network Resource Provisioning
    - GAAA-NRP and XACML-NRP profile
    - Multidomain Network Resource Provisioning (NRP) model and workflow

Use Case – e-Science infrastructure

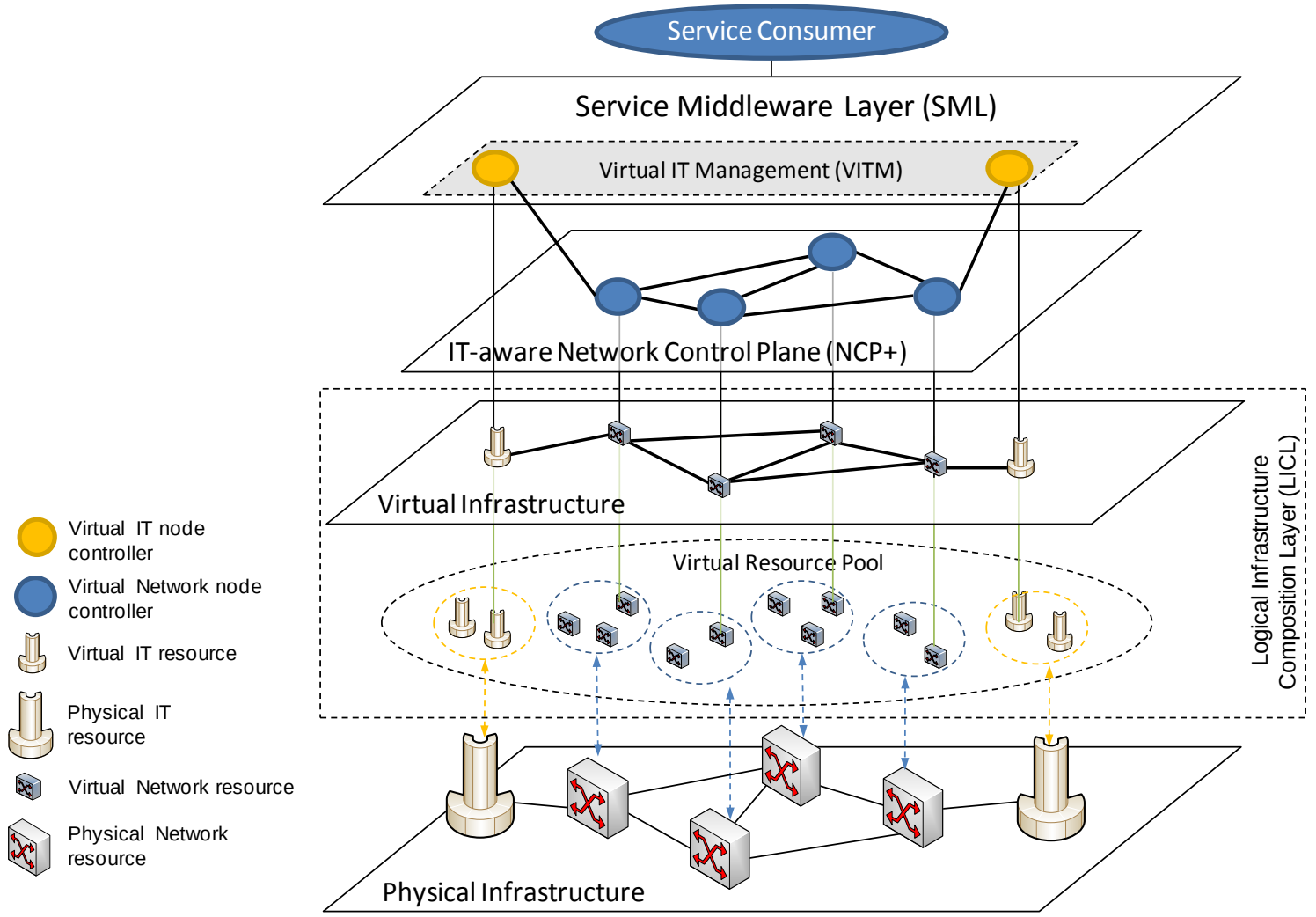


On-demand infrastructure services provisioning environment

- Security along the whole provisioning process and service/infrastructure lifecycle
- Manageable/user controlled security
- Securing remote executing environment
- Security context/session management

Components of the typical e-Science infrastructure involving multidomain and multi-tier Grid and Cloud resources and network infrastructure

GEYSERS Reference Model for Infrastructure Services Virtualisation



RORA model Roles:

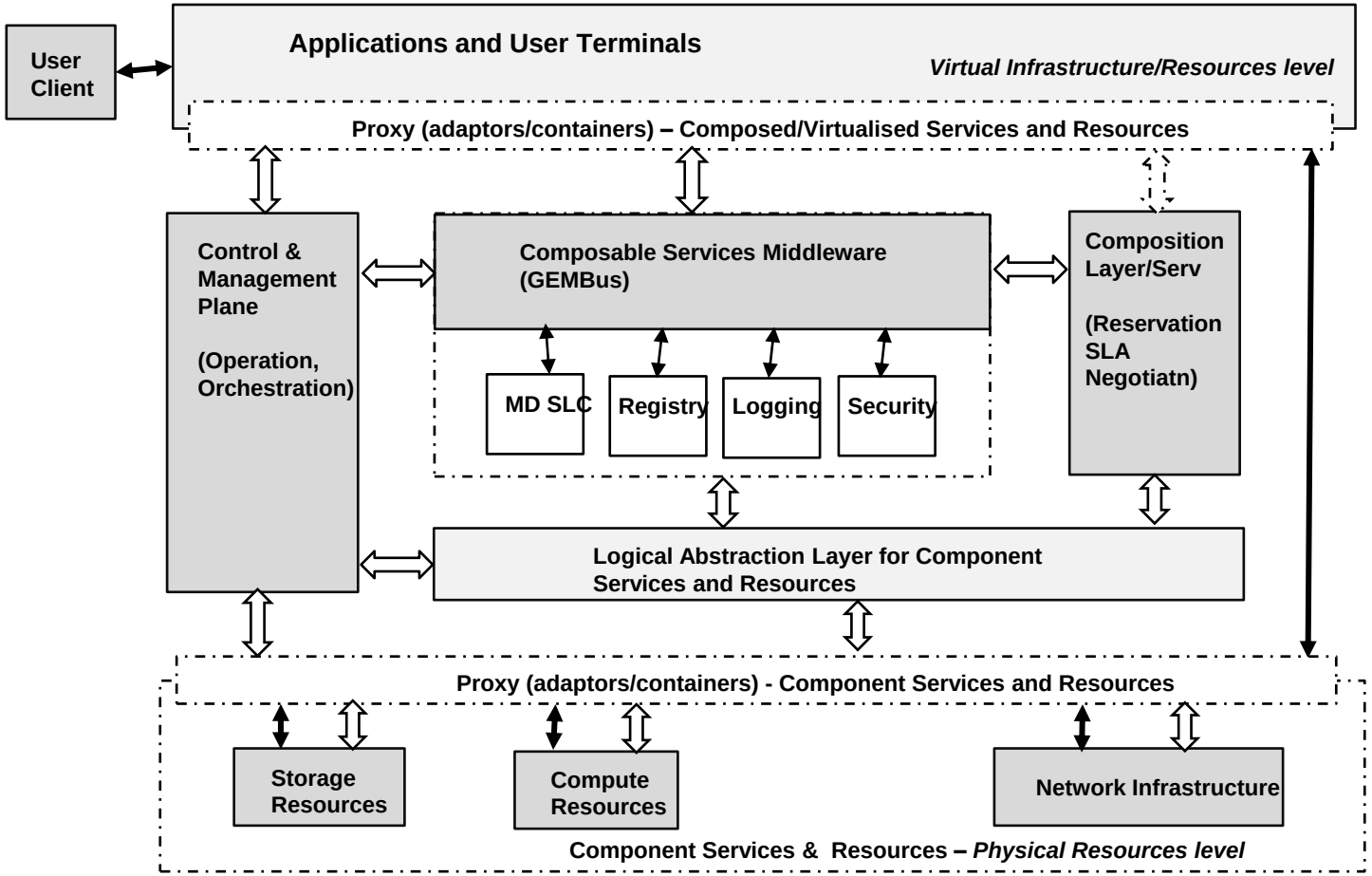
- VIO – Virtual Infrastructure Operator
- VIP - Virtual Infrastructure Provider
- PIP - Physical Infrastructure Provider

Security Service Lifecycle Management in On-Demand Resources/Services Provisioning

- On-Demand Infrastructure Services Provisioning requires definition of Services Lifecycle Management
  - Multidomain multi-provider environment
  - Includes standard virtualisation procedures and mechanisms
- Requires dynamic creation of Security/Trust Federations in multi-domain environment
  - Based on available Trust Anchors
    - Physical Resources (hosting platforms)
    - SLA or SLA negotiators/contractors
    - All other security context/credentials/keys should be derived from them
- Access control infrastructure dynamically created and policy/attributes dynamically configured
  - Access/authorisation session/context management

- ***Composable Services Architecture (CSA) as a platform for dynamically configurable composable services provisioning***

Composable Services Architecture



Composable Services lifecycle/provisioning stages

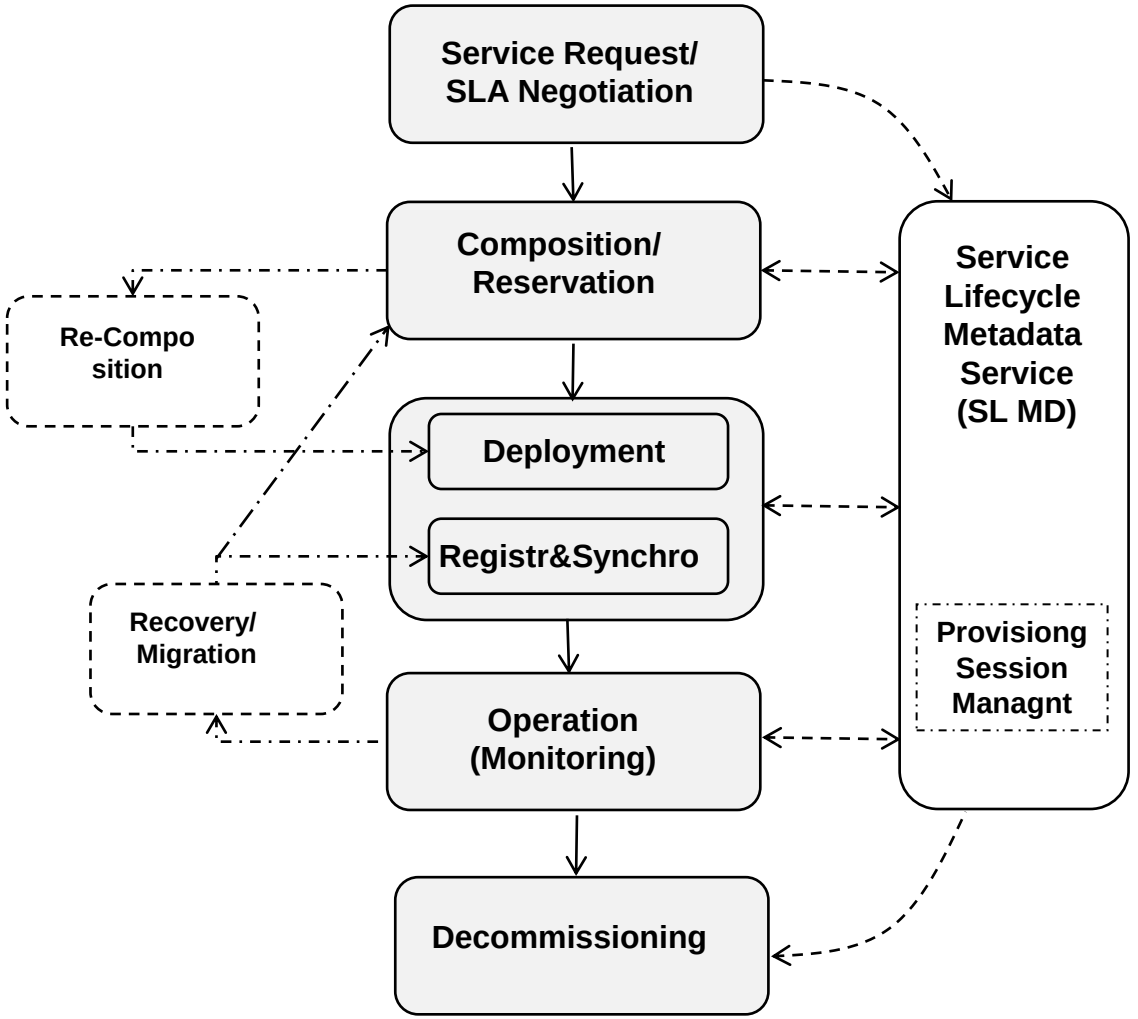
(1) Request
(2) Composition/ Reservation
(3) Deployment
(4) Operation
(5) Decommissioning

Separation of Data Plane, Control Plane, Management Plane

Control/ Mngnt Links

Data Links

CSA Services Delivery Framework (SDF)
Composable Services Provisioning Workflow



Main stages/phases

- Service Request (including SLA negotiation)
- Composition/Reservation (aka design)
- Deployment, including Reqistration/Synchronisation
- Operation (including Monitoring)
- Decommissioning

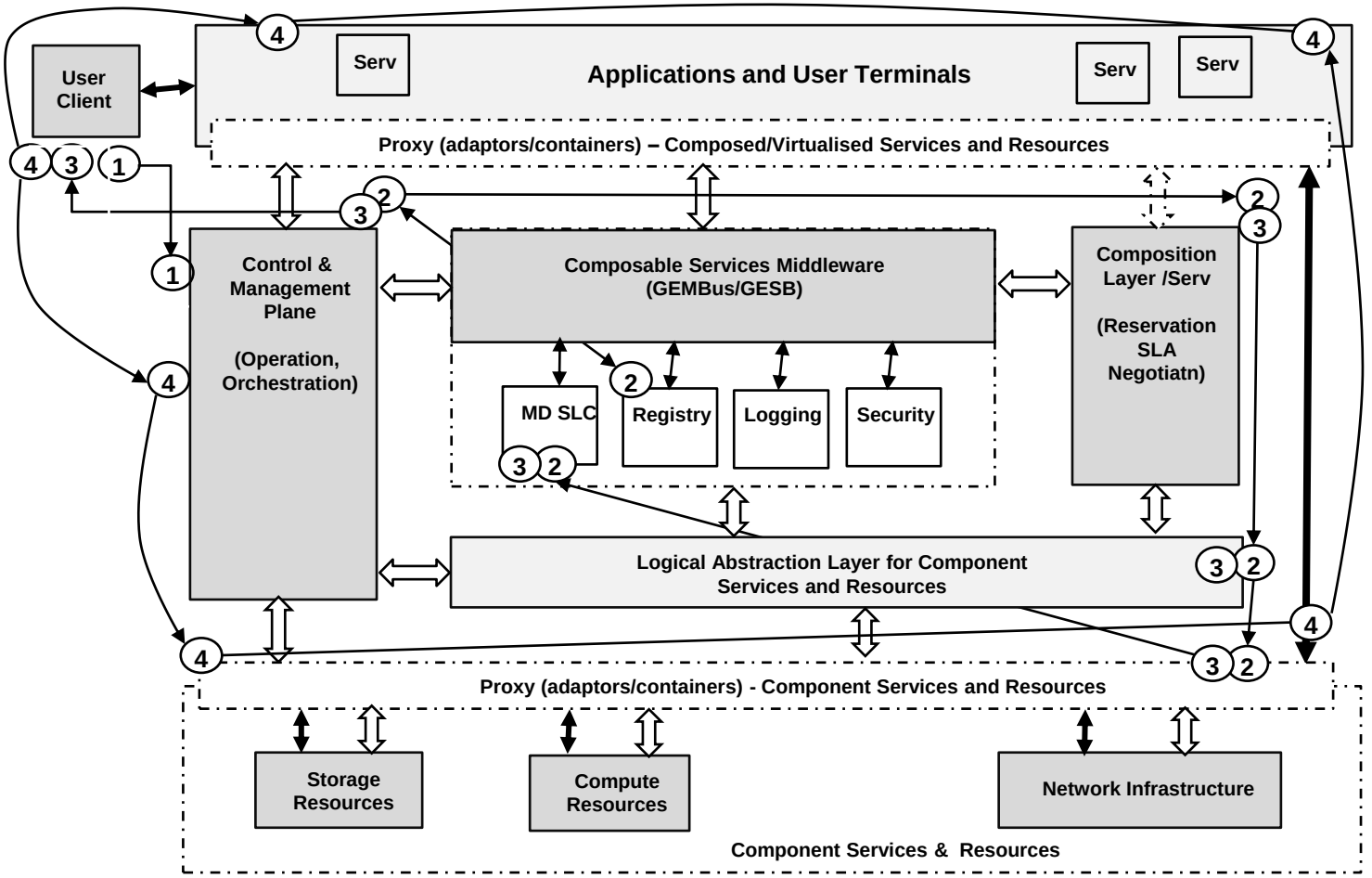
Additional stages

- Re-Composition should address incremental infrastructure changes
- Recovery/Migration can use SL-MD to initiate resources re-synchronisation but may require re-composition

The whole workflow is supported by the Service Lifecycle Metadata Service (SL MD)

Based on the TMF SDF

Composable Services Architecture – Lifecycle stages workflow



Composable Services lifecycle/provisioning stages

(1) Request
(2) Composition/ Reservation
(3) Deployment
(4) Operation
(5) Decommissioning

MD SLC – Service Lifecycle Metadata

GEMBus – GEANT Multidomain Bus

Control/ Mngnt Links

Data Links

Security Services Lifecycle Management Model (compliant to CSA SDF/lifecycle model)

Security Service request and generation of the GRI that will serve as a provisioning session identifier and will bind all other stages and related security context.

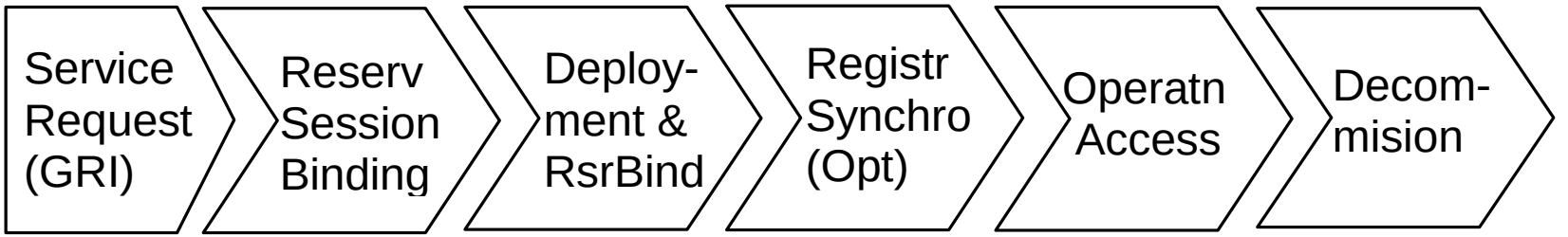
Reservation session binding that provides support for complex reservation process including required access control and policy enforcement.

Deployment stage begins after all component resources have been reserved and includes distribution of the security context and binding the reserved resources or services to GRI as a common provisioning session ID.

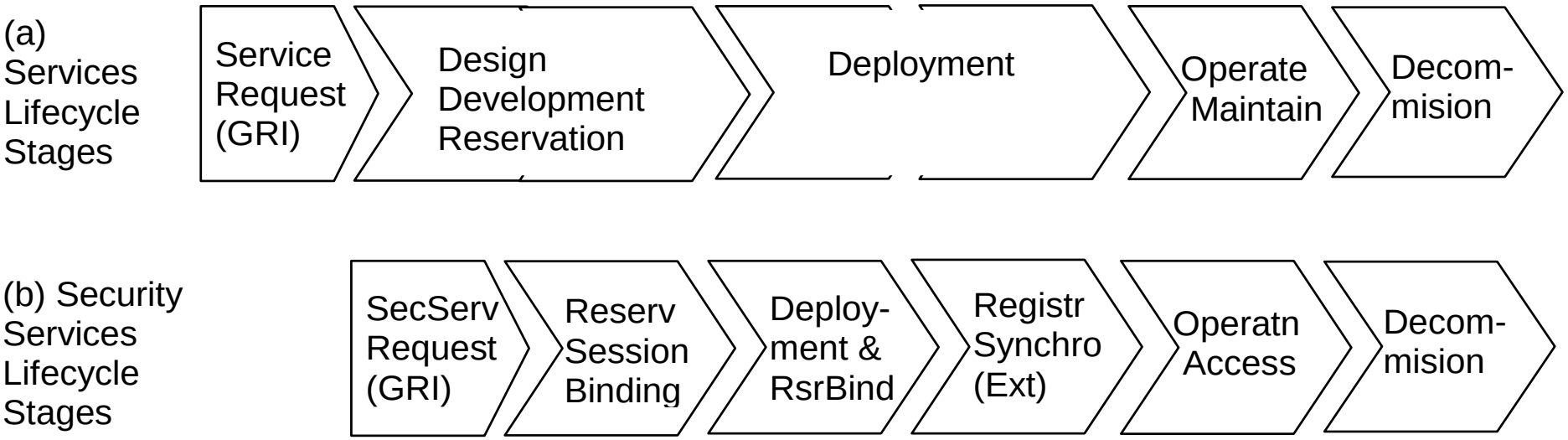
Registration&Synchronisation stage (optional) specifically targets possible scenarios with the provisioned services migration or failover/interruption. In a simple case, the Registration stage binds the local resource or hosting platform run-time process ID to the GRI as a provisioning session ID.

Operation stage - security services provide access control to the provisioned services and maintain the service access or usage session.

Decommissioning stage ensures that all sessions are terminated, data are cleaned up and session security context is recycled.



Relation between Security SLM and general SLM



Additional SSLM stages and mechanisms to ensure consistency of the security context management

Security Service Request that initiates creation of the dynamic security association and may use SLA security context.

Reservation Session Binding with GRI (as part of Planning stage) that provides support for complex reservation process including required access control and policy enforcement.

Registration&Synchronisation stage (as part Deployment stage) specifically targets possible scenarios with the provisioned services migration or failover/interruption. In a simple case, the Registration stage binds the local resource or hosting platform run-time process ID to the GRI as a provisioning session ID.

Relation between SSLM/SLM stages and supporting general and security mechanisms

SLM stages	Request	Design/Reservation Development	Deployment	Operation	Decomissioning
Process/ Activity	SLA Negotiation	Service/ Resource Composition Reservation	Composition Configuration	Orchestration/ Session Management	Logoff Accounting
Mechanisms/Methods					
SLA	**V**				**V**
Workflow		(V)		**V**	
Metadata	**V**	**V**	**V**	**V**	
Dynamic Security Associatn		(V)	**V**	**V**	
AuthZ Session Context		**V**	(V)	**V**	
Logging		(V)	(V)	**V**	**V**

Implementation suggestions

Extend existing GAAA-Toolkit pluggable Java library to support dynamic Security/AAI infrastructure creation and integration with provisioned VI

- Provides GAAA Authorisation API (GAAAPI) functions with extended AuthZ and session management functionality

Support for SDF workflow and Security Services Lifecycle Management

- Needs general infrastructure services such as Metadata SLM

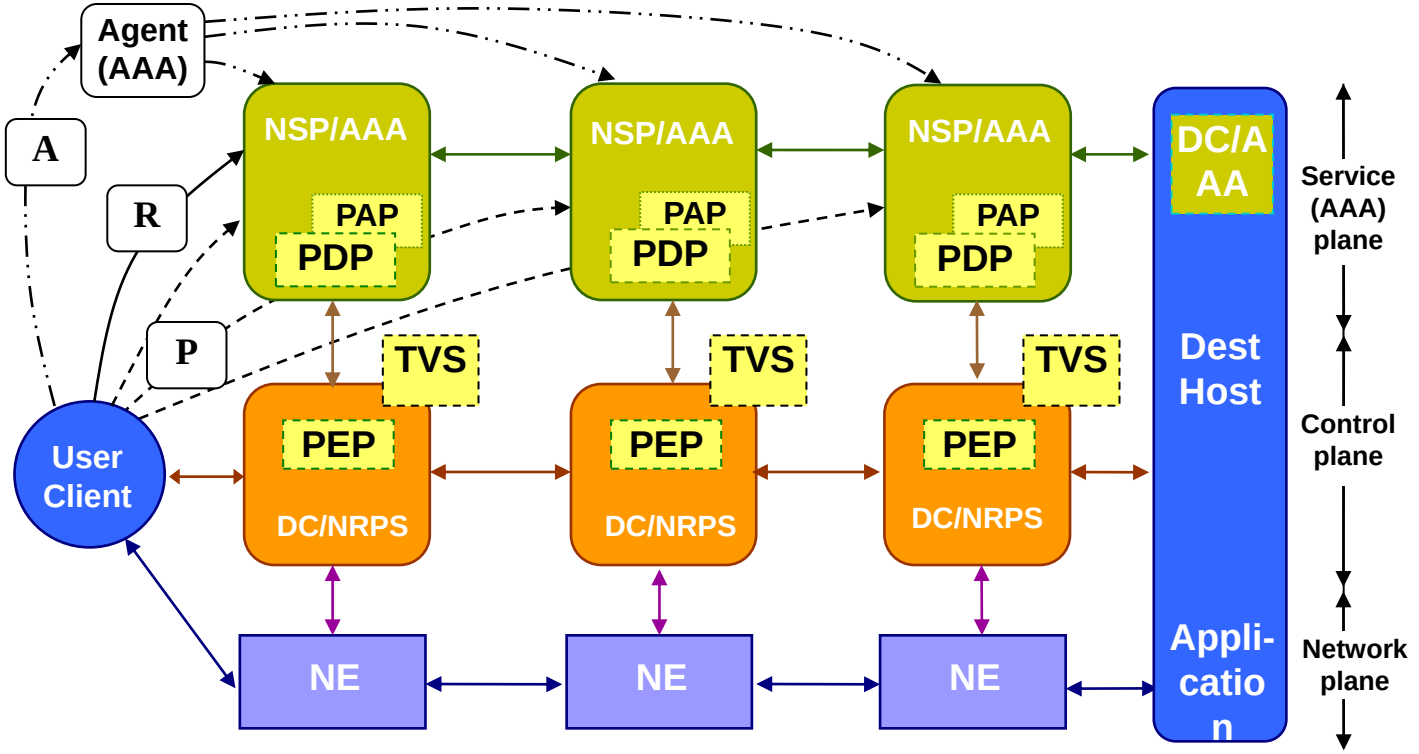
Define and implement Common Security Service Interface (CSSI)

- Supports both internal applications calls and Web service integration via Spring security
- Implements GSS-API and extends it with GAAAPI functionality

Use standard Messaging, Transport and Network security mechanisms provided by implementation platform

- Implementation platform selection – ESB/WS/SOA (Fuse, Apache ServiceMix, etc.)

Example: Multidomain Security Context Management in Network Resource Provisioning (NRP) – Provisioning sequences



Provisioning sequences

- Agent (A)
- Polling (P)
- Relay (R)

Token based policy enforcement

GRI – Global Reservation ID

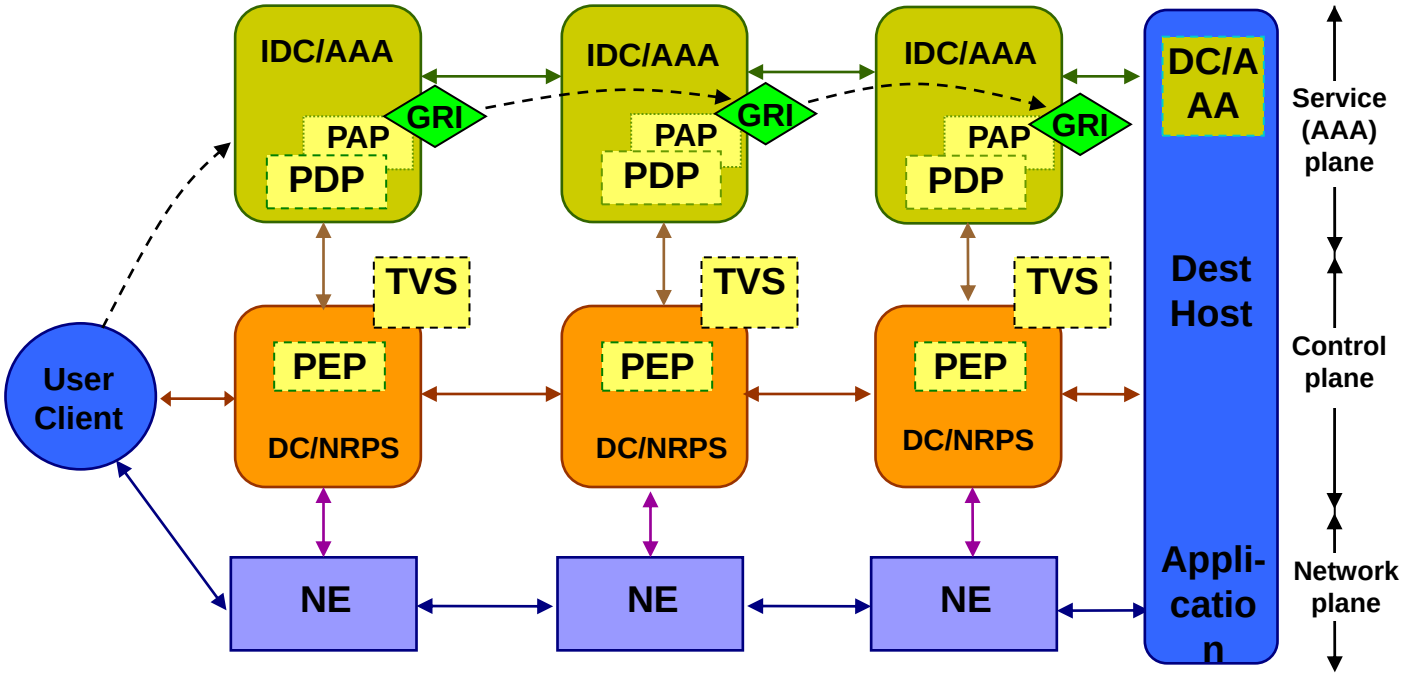
AuthZ tickets for multidomain context mngnt

T - Token

NRPS – Network Resource Provisioning System
NSP – Network Service Plain
DC – Domain Controller
IDC – Interdomain Controller

- AAA – AuthN, AuthZ, Accounting Server
- PDP – Policy Decision Point
- PEP – Policy Enforcement Point
- TVS – Token Validation Service
- KGS – Key Generation Service

Multidomain Security Context Management in NRP – Stage 1 – Path building and Advance Reservation



Token based signalling and access control

GRI – Global Reservation ID

AzTicket – AuthZ ticket for multidomain context mngnt

AT – Access Token

Pilot Token type 3 used at the Stage 1 Reservation for signalling and interdomain context communication

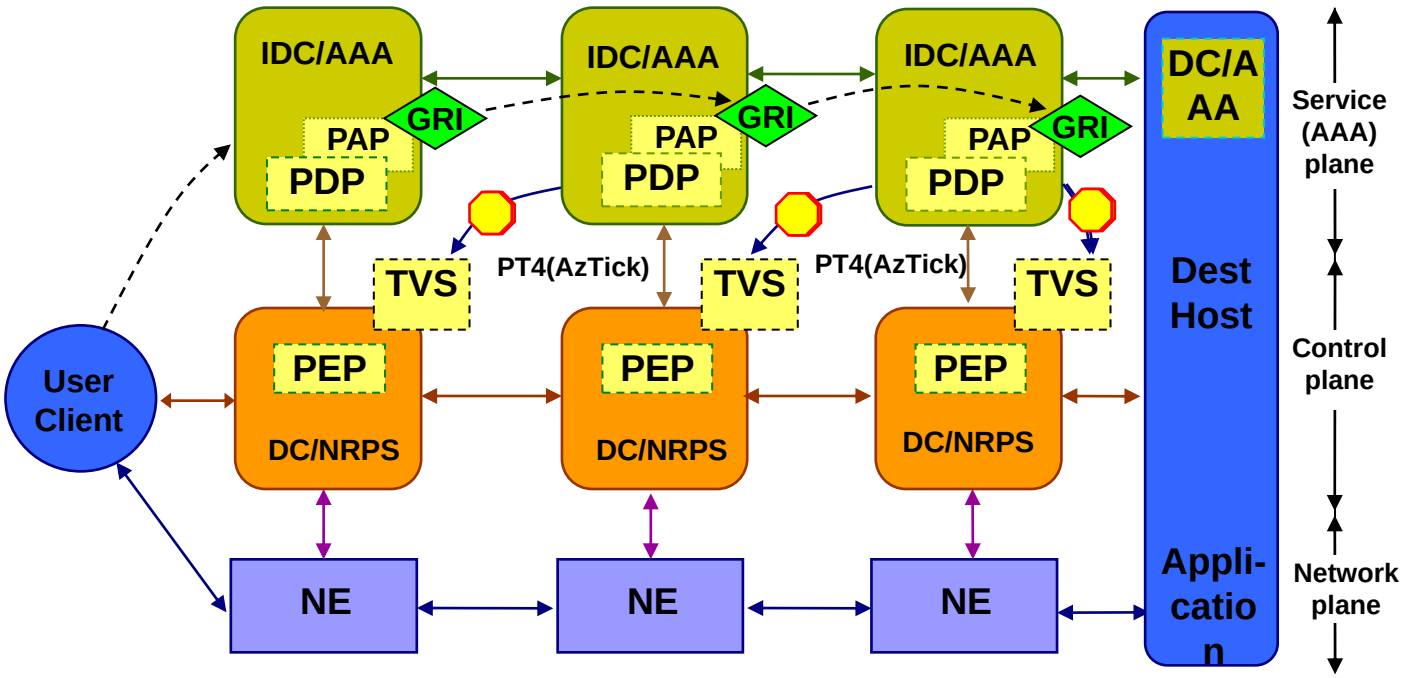
* As container for GRI and AzTicket

Pilot Token type 4 used at the Stage 2 for setup information communication

IDC – Interdomain Controller
DC – Domain Controller
NRPS – Network Resource Provisioning System
NE - Network Element

AAA – AuthN, AuthZ, Accounting Server
PDP – Policy Decision Point
PEP – Policy Enforcement Point
TVS – Token Validation Service

Multidomain Security Context Management in NRP – Stage 2 – Deployment (setup and key distribution)



Token based signalling and access control

GRI – Global Reservation ID

AzTicket – AuthZ ticket for multidomain context mngnt

AT – Access Token

Pilot Token type 3 used at the Stage 1 Reservation for signalling and interdomain context communication

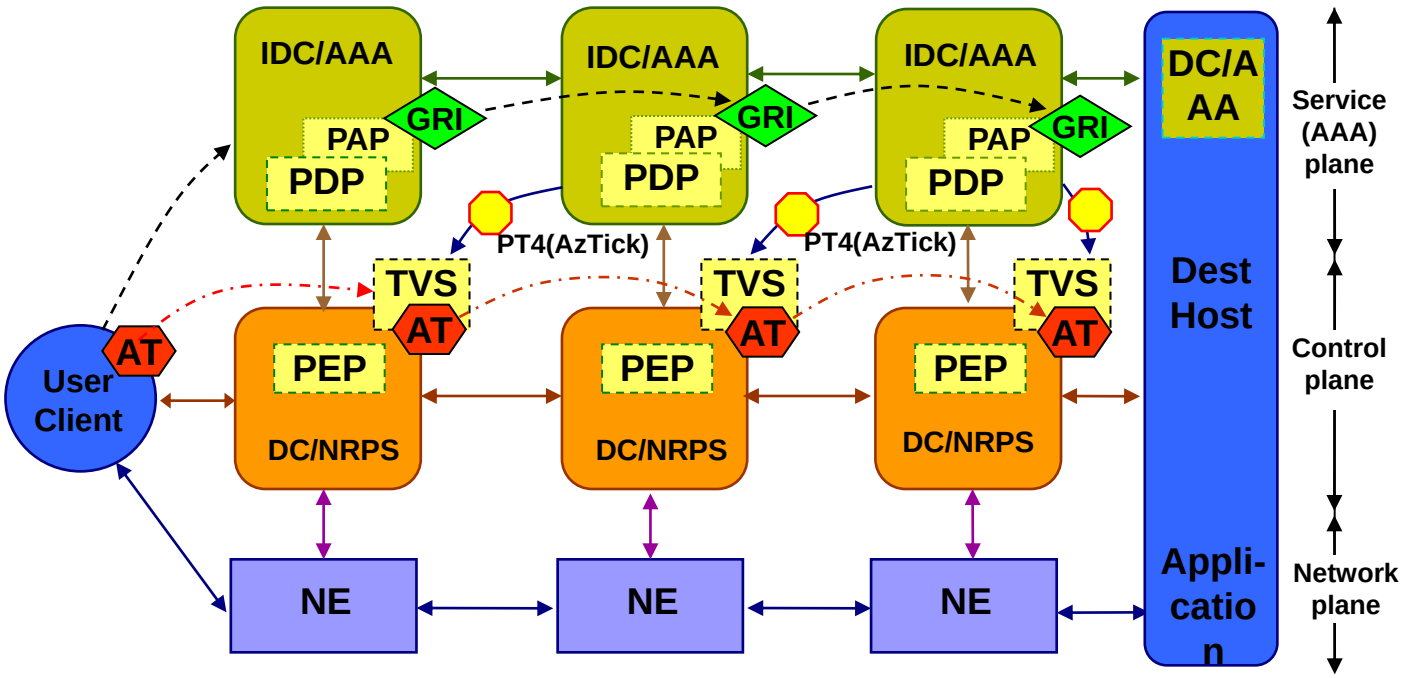
* As container for GRI and AzTicket

Pilot Token type 4 used at the Stage 2 for setup information communication

IDC – Interdomain Controller
DC – Domain Controller
NRPS – Network Resource Provisioning System
NE - Network Element

AAA – AuthN, AuthZ, Accounting Server
PDP – Policy Decision Point
PEP – Policy Enforcement Point
TVS – Token Validation Service

Multidomain Security Context Management in NRP – Stage 3 – Access Control (using access tokens)



Token based signalling and access control

GRI – Global Reservation ID

AzTicket – AuthZ ticket for multidomain context mngnt

AT – Access Token

Pilot Token type 3 used at the Stage 1 Reservation for signalling and interdomain context communication

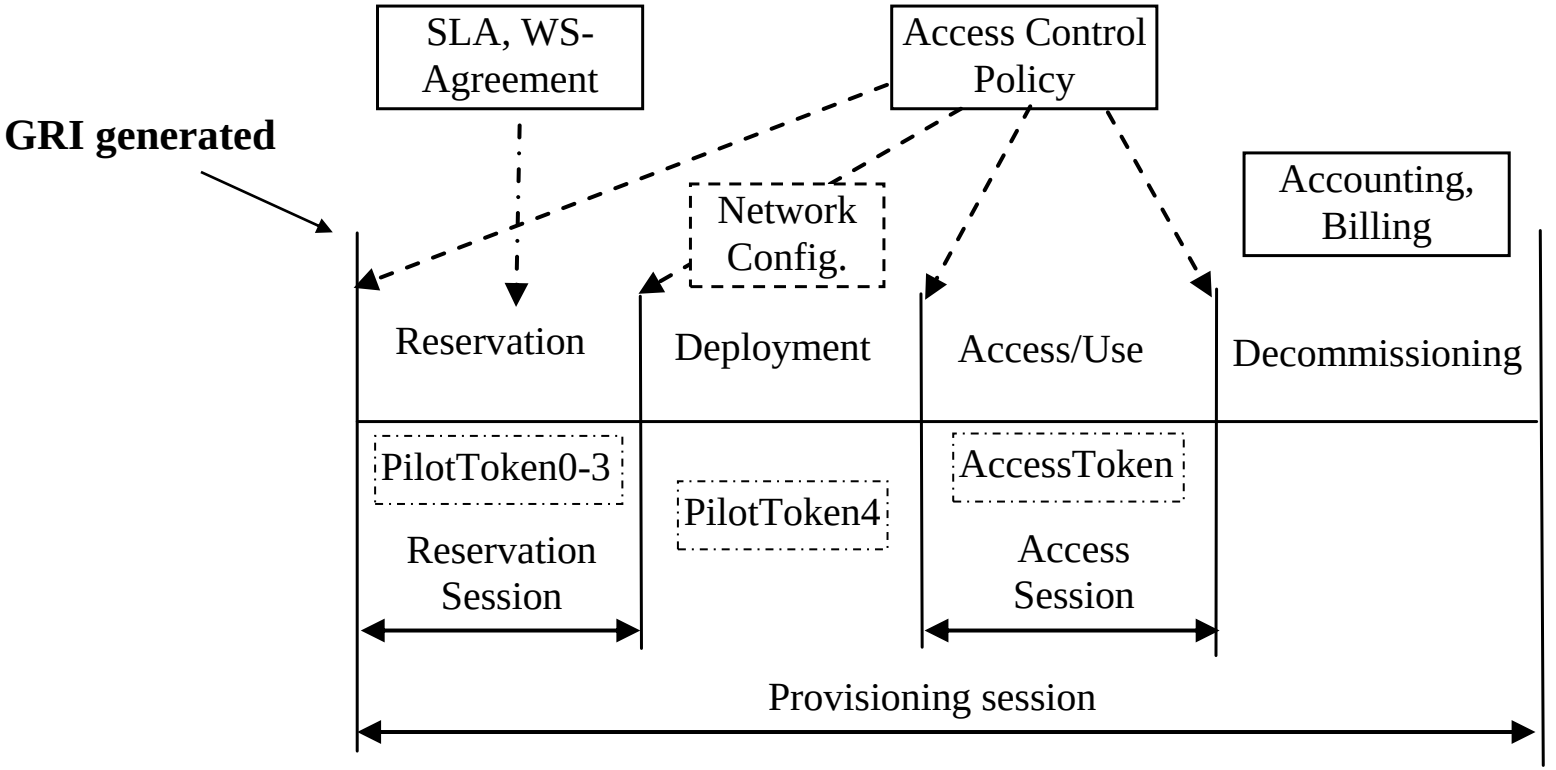
*** As container for GRI and AzTicket**

Pilot Token type 4 used at the Stage 2 for setup information communication

IDC – Interdomain Controller
DC – Domain Controller
NRPS – Network Resource Provisioning System
NE - Network Element

AAA – AuthN, AuthZ, Accounting Server
PDP – Policy Decision Point
PEP – Policy Enforcement Point
TVS – Token Validation Service

NRP Stages and Session Types



Requires consistent security and session context management

Global Reservation ID (GRI) is created at the beginning of the provisioning session (Reservation stage) and binds all sessions

Access Token and Pilot Token Types

AType 0 – Simple access token (refers to the reserved resources context)

AType 1 – Access token containing Obligations (e.g. XACML Policy Obligations) collected from previous domains

PType 0 – Container for GRI only

PType 1 – Container for communicating the GRI during the reservation stage

- Contains the mandatory SessionId=GRI attribute and an optional Condition element

PType 2 – Origin/requestor authenticating token

- TokenValue element contains a value that can be used as the authentication value for the token origin
- TokenValue may be calculated of the (GRI, IssuerId, TokenId) by applying e.g. HMAC function with the requestor's symmetric or private key.

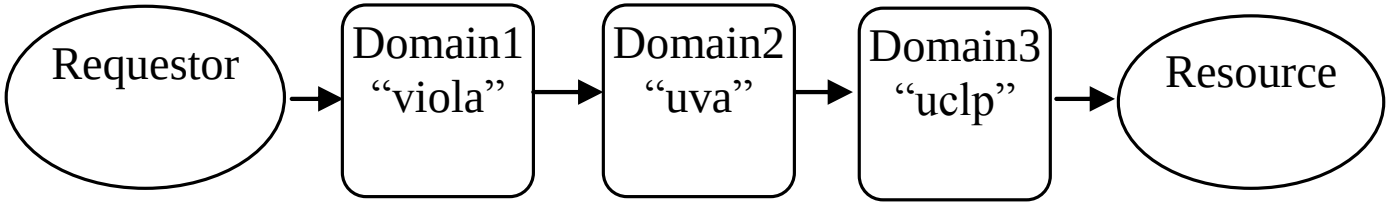
PType 3 – Extends Type 2 with the Domains element that allows collecting domains security context information when passing multiple domains during the reservation process

- Domains' context may include the previous token and the domain's trust anchor or public key

PType 4 – Used at the deployment stage and can communicate between domains security context information about all participating in the provisioned lightpath or network infrastructure resources

- Can be used for programming/setting up a TVS infrastructure for consistent access control tokens processing at the resource access stage

Chaining Pilot Tokens in multidomain signalling



- Pilot Token type 3 issued by domain UvA
- Contains SecCtx from previous Viola domain

```
<AAA:AuthzToken xmlns:AAA="http://www.aaauthreach.org/ns/AAA"
          Issuer=http://testbed.ist-phosphorus.eu/uva/AAA/TVS/tokenpilot
          SessionId="740b241e711ece3b128c97f990c282adcbf476bb"
          TokenId="dc58b505f9690692f7a6312912d0fb4c"  type="pilot-type3">
  <AAA:TokenValue>190a3c1554a500e912ea75a367c822c09eceaa2f </AAA:TokenValue>
  <AAA:Conditions NotBefore="2009-01-30T08:57:40.462Z"  NotOnOrAfter="2009-01-30T09:21:40.462Z"/>
  <AAA:DomainsContext>
    <AAA:Domain domainId="http://testbed.ist-phosphorus.eu/viola">
      <AAA:AuthzToken Issuer="http://testbed.ist-phosphorus.eu/viola/aaa/TVS/token-pilot«
            SessionId="2515ab7803a86397f3d60c670d199010aa96cb51"
            TokenId="c44a2f5f70346fdc2a2244fecbcdd244">
        <AAA:TokenValue>dee1c29719b9098b361cab4cfcd086700ca2f414
        </AAA:TokenValue>
        <AAA:Conditions NotBefore="2009-01-30T07:57:35.227Z"
                        NotOnOrAfter="2009-01-31T07:57:35.227Z"/>
      </AAA:AuthzToken>
      <AAA:KeyInfo> http://testbed.ist-phosphorus.eu/viola/_public_key_ </AAA:KeyInfo>
    </AAA:Domain>
  </AAA:DomainsContext>
</AAA:AuthzToken>
```

Future work and Discussion

- Definition of and reference implementation of the Common Security Services Interface (CSSI)
  - As extension to industry adopted GSS-API
  - Incorporate GAAA-AuthZ (RFC2904) Authorisation interface
  - Extends for Session Security Context Management and dynamic trust/security association management
- Wide range of formalisation and modeling work
- Implementation in projects GEANT3 and GEYSERS

- CSA and Security Services Lifecycle Management model is proposed as a possible deliverable for OGF ISOD RG

Acknowledgement

This work is supported by the FP7 EU funded project GEANT3 (FP7-ICT-238875), and the FP7 EU funded Integrated project The Generalised Architecture for Dynamic Infrastructure Services (GEYSERS, FP7-ICT-248657).