Multi-Domain Authorization for e-Infrastructures

Leon Gommans

Multi-Domain Authorization for e-Infrastructures

A study considering what is needed to provide authorized access to e-infrastructure network resources that are owned by multiple autonomous parties that have to trust each other to provide a service across domains

Leon Gommans

Cover painting: Betty Gommans
Cover design: Sinds1961 | grafisch ontwerp
Lay-out: Sinds1961 | grafisch ontwerp
Printed by: Print Service Ede

ISBN: 978-94-91602-26-9

Multi-Domain Authorization for e-Infrastructures

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. D.C. van den Boom
ten overstaan van een door het college voor promoties ingestelde
commissie, in het openbaar te verdedigen in de Agnietenkapel
op dinsdag 2 december 2014, te 14:00 uur

door

Leonardus Hubertus Marie Gommans

geboren te Heerlen.

Promotoren:	Prof. dr. ir. C.T.A.M de Laat
	Prof. dr. R.J. Meijer
Overige leden:	Prof. dr. ir. W. Lourens
	Prof. dr. ir. H.J. Bos
	Prof. dr. J.A. Bergstra
	Prof. dr. P.W. Adriaans
	Dr. P. Grosso

Faculteit der Natuurwetenschappen, Wiskunde en Informatica.

Opgedragen aan mijn vader en aan mijn nichtje Ellen.
Beiden zouden zo graag getuige zijn geweest.

Table of Content

Abbreviations

Term	Description
AAA	Authentication Authorization Accounting (IETF)
AAAARG	AAA Architecture Research Group (IRTF)
AAA-TSM	AAA Transaction/Session Management
AAAWG	AAA Working Group (IETF)
AC	Attribute Certificate
ACL	Access Control List
AFT	Authenticated Firewall Transversal (IETF Working Group)
AL2S	Advance Level 2 Service (Internet 2 service)
AMS-IX	Amsterdam Internet eXchange
ANSI	American National Standards Institute
API	Application Programming Interface
ARP	Address Resolution Protocol (IETF)
AS-ADF	Application Specific Authorization Decision Function
ASI	Application Specific Information
ASM	Application Specific Module.
ASON	Automatically Switched Optical Network
ASTN	Automatic Switched-Transport Network
ATM	Asynchronous Transfer Mode
AuthN	Short for Authentication
AuthZ	Short for Authorization
AVP	Attribute Value Pair
BB	Bandwidth Broker
BGP	Border Gateway Protocol (IETF)
BoD	Bandwidth on Demand
BoF	Birds of Feather meeting (the IETF way to get a working group chartered)
BonD	Bandwidth on Demand
BSA	Business Support Agent
CAT	Common Authentication Technologies (IETF Working Group)
CDR	Common Data Representation (OMG)
CERN	Conseil Européen pour la Recherche Nucléaire (European Organisation for Nuclear Research)
CHAP	Challenge-Handshake Authentication Protocol (IETF)
CIM	Common Information Model (DMTF)
CMAD	Content Monitoring and Action Device
Control Plane	Part of a network architecture that controls the behaviour of devices that forward packets.
COPS	Common Open Policy Service (IETF)
CORBA	Common Open Request Broker Architecture (OMG)

Term	Description
CRL	Certificate Revocation List
CSA	Client System Agent
DAC	Discretionary Access Control
Data Plane	Part of a network architecture that decides how to forward packets arriving on an inbound interface
DCAS	Domain Central Authorization Service
DCN	Dynamic Circuit Network (Internet 2)
DEN	Directory Enabled Networking
DiffServ	Differentiated Services (IETF)
DMTF	Desktop Management Task Force
DRAC	Dynamic Resource Allocation Controller (Nortel)
EAP	Extensible Authentication Protocol (IETF)
EASA	European Aviation Safety Agency
EGI	European Grid Initiative
EMI	European Middleware Initiative
E-NNI	External Network-to-Network Interface
e-Science	Science enable by e-technologies
FAA	Federal Aviation Administration
FCC	Federal Communications Commission (US)
FIPS	Federal Information Processing Standards (US)
FNO	Federative Network Organisation
FSM	Finite State Machine
GeGC	Global eduroam Governance Committee (eduroam)
GENI	Global Environment for Network Innovations
GFD	Grid Forum Document (Open Grid Forum document)
GGF	Global Grid Forum (became Open Grid Forum)
GHPN	Grid Hi Performance Networking (OGF working group)
GIOP	General Inter-ORB Protocol (OMG)
GLIF	Global Lambda Integrated Facility
GMPLS	Generalized Multi-Protocol Label Switching (IETF)
GNC	Global Network Carrier
GOLE	GLIF Open Lightpath Exchange
GRI	Global Reservation Identifier
GSS	Generic Security Services
GTAF	Globus Toolkit Authorization Framework
HDFS	HaDoop File System
HMAC	Hash Based Message Authentication Code
HMAC-SHA1	HMAC using Secure Hashing Algorithm – 1 (NIST)
HPC	High Performance Computing

Term	Description
HTTP	HyperText Transfer Protocol (IETF)
HTTPS	HyperText Transfer Protocol Secure (IETF)
IaaS	Infrastructure as a Service
IBC	Identity Based Cryptography
ICA	Interbank Card Association
ICAO	International Civil Aviation Organisation
ICP	Inter-Connection Point
ID	Identifier
IDC	Inter-Domain Controller
IdP	Identity Provider
IETF	Internet Engineering Task Force
Internet2	Community of U.S. and international leaders in research, academia, industry and government who create and collaborate via innovative technologies as such enabling the future or our modern digital lives.
IP	Internet Protocol (IETF Standard)
IPSEC	Internet Protocol SECurity (IETF Working Group)
IRTF	Internet Research Task Force
ISO	International Organisation for Standardization (IOS)
ISOC	Internet Society (parent organisation of the IETF/IRTF)
ISP	Internet Service Provider
ISS/VIOLA	Intelligent grid Scheduling System / VIOLA meta-scheduler system
ISSLL	Integrated Service over Specific Link Layers (IETF)
IT	Information Technology
ITU-T	International Telecommunications Union-Telecommunications sector
IXP	Internet eXchange Point
JiT	Just in Time Services
LAR	Link Access Request
LAS	Link Access Service
LDAP	Light-weight Directory Access Protocol (IETF)
LHC	Large Hadron Collider
LHCOPN	LHC Optical Private Network
Lightpath	A dedicated network connection provisioned using optical network technologies that provides a guaranteed bandwidth. A lightpath may span multiple provider domains.
LRS	Link Request Service
LSR	Label Switch Router (IETF)

Term	Description
MAC	Mandatory Access Control (authorization)
MAC	Message Authentication Code (cryptography)
MAC address	Media Access Control address (part of IEEE 802.3 standard)
MC	MasterCard
MEMS	Micro Electro-Mechanical System
MERIT	Michigan Educational Research Information Triad
MPLS	Multi-Protocol Label Switching (IETF)
NAI	Network Access Identifier (e.g. student@university.edu)
NAS	Network Access Server (IETF)
NFV	Network Functions Virtualisation
NIST	National Institute of Standards and Technology (US)
NMS	Network Management System
NPG	Network Provider Group
NPU	Network Processor Unit
NREN	National Research and Education Network
NRN	National Research Network (short for NREN)
NSA	Network Service Agent
NSI	Network Service Interface
NTP	Network Time Protocol (IETF)
OASIS	Organisation for the Advancement of Structured Information Standards
OCCI	Open Cloud Computing Interface (OGF)
OGF	Open Grid Forum
OMB	Office of Management and Budget (US government)
OME/HDXc	Nortel OME 6500 & HDXc SONET/SDH Optical cross connect switches
OMG	Object Management Group
ONF	Open Network Foundation
OPN	Optical Private Network
ORB	Object Request Broker (OMG)
OSI	Open Systems Interconnect
OSS	Operations Support System
OVPN	Optical Virtual Private Networks
PAP	Password Authentication Protocol (IETF)
PDP	Policy Decision Point (IETF)
PDU	Protocol Data Units
PEP	Policy Enforcement Point (IETF)
PFWG	Policy Framework Working Group (IETF)
PIN	Personal Identification Number

Term	Description
PIP	Policy Information Point (IETF)
PKC	Public Key Certificate (IETF)
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure X.509 (IETF Working Group)
PMI	Privilege Management Infrastructure
PoS	Point of Sale
PPP	Point-to-Point Protocol (IETF)
PRP	Policy Retrieval Point (IETF)
PXC	Photonic Cross Connect
QoS	Quality of Service.
RADIUS	Remote Authentication Dial In User Service (IETF)
RAP	Resource Allocation Protocol (IETF Working Group)
RBAC	Role Based Access Control
RC	Roaming Confederation (eduroam)
RFC	Request For Comment (IETF document)
RM	Resource Manager
RO	Roaming Operator (eduroam)
RSVP	ReSerVation Protocol (IETF)
RSVP-TE	ReSerVation Protocol - Traffic Engineering (IETF)
SAML	Security Assertion Markup Language (OASIS)
SAP	Service Access Point
SC	SuperComputing (annual super-computing technology event)
SDH	Synchronous Digital Hierarchy (Telco Carrier Network protocol)
SDN	Software Defined Networking
Service Plane	Part of a network architecture that has awareness of network service to be delivered and understands how to provision such service in the network.
SIP	Session Initiation Protocol (IETF)
SIP/DIP	Source IP / Destination IP address pair (IETF)
SML	Service Management Layer
SOAP	Simple Object Access Protocol (W3C standard)
SOCKS	SOCKet Secure, protocol allowing control of network firewall functions (IETF)
SONET	Synchronous Optical Networking (Telco Carrier Network protocol)
SP	Service Provider
SPG	Service Provider Group
SSH	Secure Shell (IETF)
SSL	Secure Socket Layer (IETF)

Term	Description
TBS	Token Based Switch
TCP	Transport Control Protocol (IETF)
TL-1	Transaction Language -1
TLS	Transport Layer Security (IETF)
TMN	Telecommunications Management Network
TRA	Token Request Authority
TSM	Transport/Session Management
TVS	Token Validation Service
UCLP	User-Controlled LightPath (Canarie)
UHO	User Home Organisation
VLAN	Virtual Local Area Network
VLSR	Virtual Label Switch Router
VM	Virtual Machine
VMTC	Virtual Machine Traffic Controller
VO	Virtual Organisation
VOMS	Virtual Organisation Membership Service
VPN	Virtual Private Network
W3C	World Wide Web Consortium
WADA	World Anti-Doping Agency
WG	Working Group (of a standards body)
WS-Trust	WebService Trust (OASIS)
X.509	ITU-T standard for Public Key Infrastructure and Privilege Management Infrastructure
XACML	eXtensible Access Control Markup Language (OASIS)
XEN	Open Source VM Hypervisor

Note to the reader:

As many sections are based on publications, the reader may notice a mix of both UK and US spelling conventions, as material was published by US and UK organisations.

1 Introduction

"Begin with the end in mind."

Stephen Covey (1932-2012)
Educator, author, businessman and keynote speaker

1 Introduction

1.1 Introduction

The overall objective of this thesis is to study what is needed to create a multi-domain authorization system, allowing applications to access e-Infrastructure resources. Typically, resources that contribute to an e-Infrastructure are owned by multiple service providers. The need for a multi-domain authorization system emerges when access control to such resources needs to be automated. The study contributes to the understanding of what is needed by defining two frameworks and an authorization architecture. The first framework provides a way to articulate authorization scenarios; the second framework helps to understand the role of trust within authorization systems. A generic authorization architecture was defined as a way to help guide the solution design of an authorization system. The generic architecture was validated in collaboration with pioneering Internet research organisations to show its applicability. The research on the second framework, presented in chapter 5, indicates that the complexity of an authorization system can decrease if organisational trust and power is considered along with its design, allowing the use of simple tokens.

The need to consider the objective of this thesis emerged in 1998 within the Computational Physics Group at University of Utrecht. The group collaborated with FOM Rijnhuizen in performing remote experiments with a plasma fusion reactor located at the Kernforschungsanlage (Nuclear Research Facility) in Jülich, Germany. The experiments required a dedicated, high capacity network connection between both institutes. Providing network connections for scientific research is a responsibility of National Research and Education Networks (NRENs). NRENs must collaborate to provide such network connections across countries. At that time, network technologies would allow automated creation of dedicated bandwidth connections within each NREN. However, automating the creation of an end-to-end chain of connections across network "kingdoms", proved to be a challenge. Fig 1.1 illustrates the problem we started to call the "multi-kingdom" problem.

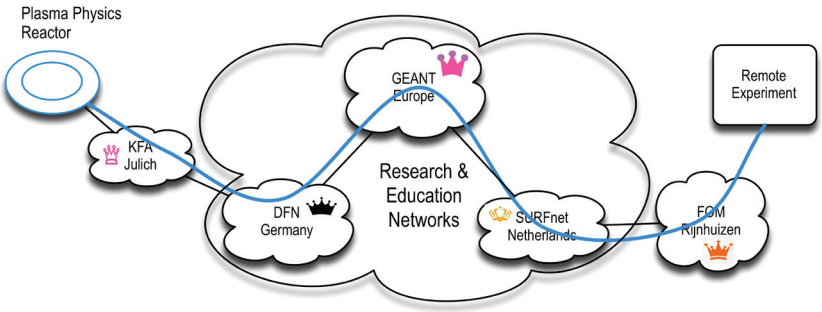


Fig. 1.1: The problem of chaining dedicated network connections across multiple network "kingdoms" of participating institutes and National Research and Education Networks initiated the study into what is needed to allow such network connections to be created and linked automatically.

It was recognized that the lack of authorization capabilities formed an important underlying issue. Before institutes or NRENs can allow automated access to its high value network assets, these organisations would first need to have the ability to implement policies governing such access. Each autonomous NREN is required to permit or deny such requests based on its own policies, whilst being able to provide correctly interoperating network resources within the chain. It was thought that a study on what is needed to create such an authorization system, capable of handling access request transactions across multiple NREN domains automatically, would be an important contribution to help fill such a gap.

The role of multi-domain authorization, supporting scientific applications, became more important. In the late 90s it was observed that the Internet started to connect an increasing variety of resources that, when chained together, offered additional value to a scientific user. From examples such as high-energy physics [CERN], radio astronomy (fig 1.2) [ASTRON, ELVBI], visualisation [EVL], etc. initiatives emerged to create large e-Science infrastructures, which became known as e-Infrastructures [EIRG]. With such forefront type of developments, we saw a growing need to globally share large amounts of data, advanced instruments, processing-, network-, visualisation capabilities and the like. It was envisaged that such infrastructures would need authorization systems, allowing their owners to define policies controlling resources access. In our minds, as it did in the first example, a study on understanding what is needed to create multi-domain authorization systems would contribute to the evolution of e-Infrastructures. In e-Infrastructures, NRENs play a key role in providing connections and access. Therefore, our original question evolved into the question what is needed to create a multi-domain authorization system for e-Infrastructures. We subsequently started to look for technologies that could resolve that problem.

Fig 1.2 The European array of radio telescopes that require dedicated network connections allowing real-time correlation of signals from different telescopes to perform observations (Very Long Baseline Interferometry).

One can imagine that multi-domain authorization is an old problem appearing in many different cases. Over time, the problem has been solved in many ways using various technologies. This thesis will show that authorization systems have used technologies ranging from clay tokens in sealed envelopes, during the Neolithic era, to electronic transactions, for example to authorize a payment in the modern Internet era. In our e-Infrastructure context, dedicated network connections are needed by applications that transport massive amounts of data, as they would otherwise disturb the regular Internet [DLA3]. At the start of the research, chaining dedicated network connections across multiple NRENs was performed by hand. After each organisation approved its use, network switches were configured and cables were plugged to correctly route connections. Such a process would take several weeks, hundreds of emails and many phone calls by the requestor and the participating NRENs before a connection became operative. To allow scientific applications automated access to such network connections, it was envisioned that authorization functions would need to work alongside functions that can find and configure a suitable path, reserve resources, monitor line and equipment utilization and more. Authorization functions handle transactions that request and approve the delivery of each NREN's contribution. The initial research, therefore, focussed on the question what kind of authorization architectures, functions and technologies would be needed along with these other functions and technologies. During the research it was recognized that trust is an important factor related to the information needed to be exchanged within an authorization system. The research questioned whether a simple token can be used to refer to a requested service, reducing the need to exchange complex lists of securely asserted attributes. What is needed to arrange trust, going hand in hand with power, was studied by looking at an example from the card payment industry. During the course of the research, a number of existing typical "multi-kingdom" cases were considered to search for applicable technologies and what role trust has when applying technologies to solve the problem. It was concluded that networks can be designed to transmit token labelled data-packets or carry tokens in signalling messages. Such technologies transform a network from a transmission infrastructure to a customizable infrastructure supporting services. In this case, these services comprised of network Quality of Services that can be dedicated to single applications. This thesis shows that these services can be coupled, using an authorization system enabling the creation of end-to-end connections, by fulfilling the requirement that each domain wants to make individual decisions regarding the use of its service.

As the level of automation of IT infrastructures is continuously increasing, the future of this type of research will become more important with developments such as Software Definable Networking [SDN]. Organisations such as the Open Networking Foundation [ONF] are considering optical transport networks [OTWG] allowing its programmatic control. Network Functions Virtualisation [NFV] is a technology that allows programmatic control of virtualized network appliances such as routers, firewalls, load-balancers, gateways, etc. Grouping such functions into services will require authorization and trust aspects to be considered, in particular when functions belong to different owners. Therefore, such developments represent an important future context for additional research into the applicability of the authorization- and trust framework and the supporting Generic AAA Architecture, in particular when considering the role of simple tokens. This research therefore contributes to the transformation of the Internet's best effort service approach into a more predictable one, supporting individual business application needs on a global scale.

1.2 Research questions

To support the overall objective of this thesis: "To study what is needed to create a multi-domain authorization system, allowing applications to access e-Infrastructure resources", the following main research question has been defined:

> ***What generic authorization functions are needed to provide trusted, policy based access to combinations of e-Infrastructure resources that are owned by different parties?***

The research approach is aimed at finding generic authorization functions that can be applied to many different authorization scenarios, involving resources that are owned by different parties. Different parties are expected to be autonomous in the way they create policies that handle authorization transaction decisions. When combining different e-Infrastructure resources, parties must have a way to trust each other before authorization transactions can be handled.

A number of sub questions have been posed to guide the reader through the presented research.

> *1* ***What generic authorization functions can be distinguished and how do they interact?***

When considering authorization as a generic capability, it is important to recognize basic functional elements that are essential when authorizing access to a resource. To describe its architecture, it must be recognized how these elements relate to each other and to the outside world together with their design principles. This question should also include if there are typical authorization transaction sequences that can be recognized when these functional elements interact. In researching certain interactions, can conceptual interaction patterns be recognized? If so, can they be classified and can these patterns act as a framework to help describe the interaction concepts? Can such a framework be helpful in motivating a generic architectural approach, capable of handling the recognized interaction patterns?

> *2* ***What generic authorization concepts are expected to work best for classes of applications that use multi-domain network resources?***

After identifying authorization functions and different ways they work together when providing authorized access decisions, the question is what concepts work best for certain classes of applications. In the given research context, scenarios are considered that use network resources provided by multiple network domains to connect resources taking part in e-Science applications. As each collaborating domain may use different network technologies with different configuration parameters, it becomes increasingly difficult to find a common set of service parameters that will be understood by every participating domain as the number of collaborating domains increases. Each domain may have different business policies that domains may not want to expose, for example each domain may apply different policies to prefer certain requests above other requests when (pre-) allocating resources. Also, a domain may act as proxy or broker for other domains,

making the time to handle authorization transactions hard to predict. Therefore, the ability of different generic framework models to handle such cases must be explored.

3 How can we apply the generic multi-domain authorization concepts in Network QoS / Lightpath provisioning class of applications?

Given the research context of multi-domain networking that allows resource (pre-) allocation of guaranteed bandwidth, the question is how models of the Generic AAA approach can be implemented in such a way that its applicability can be observed. Network technologies allow many different ways to implement bandwidth guarantees and associated ways to manage and provide access to them. Network technologies use connectionless (packet-switched), connection-oriented (circuit-switched) or pure optical (lambda switched) approaches to forward traffic building end-to-end connections. The question becomes how the Generic AAA approaches can be implemented at different network layers. This leads to the question of how the generic multi-domain authorization concepts can be implemented for applications that need large amounts of dedicated network bandwidth.

4 What is needed to arrange trust when authorizing e-infrastructure resources?

The answer to sub-question 1 implies that policy based authorization decisions are made with the assumption that trust between their participants pre-exists. This research question considers this assumption in more detail, both from a technical protocol (security) perspective and from the business perspective. With a focus on the business perspective, the requirements for making policy based authorization decisions trusted by the participants involved in a transaction must be considered. The existing world already knows many such examples that have implemented trusted ways to handle authorization transactions. By considering examples taken from two extreme cases, the MasterCard system on the one side and the scientific optical networking world on the other, the question as to what is needed to arrange trust was investigated. Can these needs be captured in a framework? How is the framework expected to work in the e-Infrastructure world? Once developed, it should be possible to take another example and be able to roughly verify the framework. Consequently, the question arises what could work as the next steps allowing network e-Infrastructures, such as the Global Lambda Integrated Facility (GLIF), to scale up.

1.3 Thesis Outline

This thesis consists of 8 chapters, following a structure that can be subdivided into three research phases, providing answers to the research questions:

Phase 1 between 1998 and 2001: Generic AAA conceptual research
Phase 2 between 2001 and 2008: Generic AAA applicability research
Phase 3 between 2010 and 2013: Trust concept research.

Chapter 2 provides a historic perspective and recognises some key elements of authorization systems by considering an early example taken from pre-history. This example underlines the importance of sharing knowledge about the correct handling of authorization decisions, represented by clay tokens. Tokens, and their correct handling by putting them into sealed envelopes, created trust between community members. Subsequently, chapter 2 provides an overview of the concepts related to authorization, which were found at the Internet Engineering Task Force (IETF) and related efforts at the start of phase 1. This overview indicates that these concepts could only solve parts of our "multi-kingdom" problem. This fact helped to motivate the IETF to allow research work to be performed within an Internet Research Task Force (IRTF) group. We will then describe our IRTF work on a conceptual framework and a generic architecture for policy based authorization of decision-making functions. The functions authorize a user, requesting access to a resource, based on policy decisions. Both single- and multi-domain scenarios will be considered. The relationships between functions and the underlying transaction message sequences are used to provide a framework that classifies sequence models and provides a vocabulary to describe them. Chapter 2 then presents an architecture allowing authorisation policy decisions to take place in a decentralized way. The architecture uses a fundamental principle of separating the decision making process from handling the meaning of decisions. It shows in more detail what functional elements and protocols might be needed to handle policy based authorization decisions The work of phase 1 will provide answers to sub question 1: *What generic authorization functions can be distinguished and how do they interact?* Chapter 2 will also lead to the additional research sub questions, considering the need to validate our concepts and further considering the importance of trust. Section 2.5 will detail the research phases and explain the evolution of our research.

Chapter 3 places the concepts of chapter 2 into the multi-domain e-Infrastructure context. We will consider applying the Generic AAA Architecture concepts and the identified Authorization Framework sequence models. The thesis will show how distributed policy based authorization decision-making can work, considering the requirement that every domain should be autonomously capable of defining its own policies. It will describe several ways to contemplate solutions using the conceptual models and functional concepts. It explains which concepts are expected to work and what advantages and disadvantages certain approaches have. Furthermore, it will explain that the use of a meaningless token is a promising concept to implement policy-based decisions that combine access to pre-allocated resources owned by multiple domains. The work of chapter 3 was performed in phase 2 of the research and provides answers to sub question 2: *What generic authorization concepts are expected to work best for classes of applications that use multi-domain network resources?*

Chapter 4 describes proof of concept experiments performed with different models described by the Authorization Framework and Generic AAA Architecture, and shows how these can be applied in practice. The experiments were performed in the context of network QoS and Lightpath provisioning application scenarios, involving both single and multi-domain cases. It will demonstrate the use of two framework sequences (Agent and Push) and their combination. Generic AAA Architecture components were used to handle authorization sequences with

enforcement implemented at different network layers. The work of chapter 4 was performed in phase 2 of the research and provides answers to sub question 3: *How can we apply the generic multi-domain authorization concepts in Network QoS / Lightpath provisioning class of applications?*

Chapter 5 has been motivated by an observation made during the work in Phase 1 on the Authorization Framework: authorization cannot take place without the necessary trust being in place. After defining the concepts around trust, this chapter will present a framework constructed from observing two existing multi-domain cases; a third case is used to roughly verify its applicability. The work of chapter 5 was performed in phase 3 of the research and will answer sub question 4: *What is needed to arrange trust when authorizing e-infrastructure resources?*

Chapter 6 provides a summary of the answers to the research questions to reach the conclusions, and chapter 7 describes possible future directions. Chapter 8 provides an overview of the scientific contributions made and shows follow-up work that refers our work.

Generic AAA concepts

2

"Architecture begins where engineering ends."

Walter Gropius (1883-1969)
Architect

2 Generic AAA concepts

A pre-historic example will introduce the essence of authorization and the importance to have correct knowledge about its rules to create trust in section 2.1. After introducing the origin of the AAA (Authentication, Authorization and Accounting) acronym, we will then consider a number of concepts and technologies related to authorization that we found being worked at in the IETF when we took our multi-kingdom problem to this community. We examined the usefulness of these concepts in helping to resolve our problem. This work raised a number of additional questions that helped to refine our approach, which resulted in the research performed in IETF/ IRTF context.

In sections 2.2 and 2.3 we will explain the basic authorization concepts that were defined as result of research work performed in the IRTF AAA Architecture Research Group [AAAARG]. The goal of this Research Group was to focus on architecture supporting AAA services that:

- *can inter-operate across organizational boundaries*
- *are extensible yet common across a wide variety of Internet services*
- *enables a concept of an AAA transaction spanning many stakeholders*
- *provides application independent session management mechanisms*
- *contains strong security mechanisms that are tuned to local policies*
- *is scalable to the size of the global Internet*

Within this context, the work presented in these sections will focus on *enabling a concept of an AAA transaction spanning many stakeholders*, such that *interoperability across organizational boundaries* can be achieved. Section 2.2 will explain the AAA Authorization Framework [R2904]. A framework describing entities involved in different authorization transaction sequences, recognizing their communication pattern and defining a vocabulary to describe them. Section 2.3 will explain Generic AAA Architecture [R2903]. An architecture defining functions and relationships to handle AAA transactions in a distributed way. Generic functions include policies driven rule based engines and application specific modules that can act in a network.

Although not further described here, the cases that stood example for exploring the applicability of the framework and architecture in scenario's are described in RFC2905 "AAA Authorization Application Examples" [R2905]. This document shows the possible applicability of the Generic AAA framework and architecture for both networking applications (Roaming, Mobile IP, Bandwidth Brokerage, Internet Printing) and non-networking applications (e-Commerce and Computer based Education and Distance learning). Also essential requirements for an authorization system were derived and listed in RFC2906 "AAA Authorization Requirements" [R2906].

The work presented in section 2.2 and 2.3 evolved and matured within the Authorization Frameworks and Mechanisms Working Group of the Open Grid Forum [GFD38] and research performed within the Systems and Network Engineering group at University of Amsterdam [SNE]. Additional conceptual insights gain have been included, in particular in section 2.2.

2.1 Authorization from different perspectives

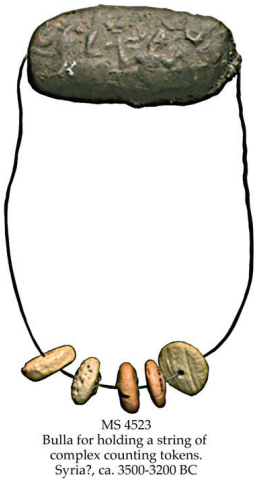
Authorization of resources used many different technologies over the ages, but the essence of authorization has remained the same: Authorization mechanisms ensure that policies, govern access to resources, are performed according to the rules established by individuals or communities. To illustrate this fact, we have used an example from pre-history. We continue with some history around the origin of AAA and then explain concepts that relate to authorization in the Internet era.

2.1.1 A pre-historic perspective

The first community based economies emerged in the Neolithic era, the pre-historic timeframe from approximately 10.000 BC ending between 4500 and 2000 BC depending on the region [HOUR]. During this period, the lifestyle of human culture moved from *"hunting and gathering"* to one of living in settlements [BOCA]. Settlements emerged around the human ability to produce wealth from using its surrounding infrastructure. By cultivating land, farming animals, finding and working natural resources [GUIS], labour and resources contributed value to the community. Sophistication, such as irrigation in agriculture, increased the ability to produce surplus yields that could be stocked. Trading economies emerged around the production and distribution of products [NEOW]. Goods and services were bartered arranging the exchange of its ownership. Neighbouring settlements started to form networks and interacted with other networks. Ever since, creation of wealth within communities was in need of a system that arranged contribution and distribution in a way that secured the interest of involved parties in a trusted way.

Based on work of Pierre Amet [AMET] and Maurice Lambert [LAM], Denise Schmandt-Besserat explains in her book *"Before Writing, from counting to cuneiform"* [SCHM, ZIMA] that tokens were used as part of transaction mechanisms involved in the exchange of goods and services in the Mesopotamian area. Tokens, made from clay, had different shapes like cones, disk, spheres, etc. Schmandt-Besserat asserts that each shape represented different items such as a sheep, a bowl of cereal, a day of labour, etc. The significance of these tokens *"as an operational device in Mesopotamian bureaucracy,"* was described by A. Leo Oppenheimer [OPPE]. This became possible when cuneiform writings on tablets were found, allowing the token system to be studied in more detail. Oppenheimer describes that tokens were used to perform accounting by authorities. Tokens were enclosed in clay envelopes to represent transactions (fig. 2.1.1). Alternatively, tokens were stringed together, securing its knot by a clay bulla (fig. 2.1.2). Both envelopes and bulla's were impressed with patterns from a seal (fig 2.1.3). As such, the process of containing and securing tokens using impressions of a seal represented the outcome of a transaction. Oppenheimer suggests that such transactions could take place for example between administrative offices of authorities that were keeping account of their resources. Within this context, the act of using a seal implies authority. The result could subsequently be stored for future reference. J.N Postgate explains in his book *"Early Mesopotamia, society and economy at the dawn of history"* [POST] that cities collaborated by using seals, with different symbols for each participating city, as a means to imply

such authority (fig 2.1.4). Postgate follows the explanation by Th. Jacobsen that such seals were used in the context of good delivery to a common stock, created for a common purpose by individual contributions from the cities collectively sealing. As such, these types of processes - using a symbol system based on tokens, envelopes, bulla's and seals - represented an early form of an authorization mechanism formally arranging contribution and re-distribution of goods. Contributions were recognized as being authorized when evaluated to be in accordance with community-established policies. Fig. 2.1.1 and 2.1.2 could represent the outcome of a barter policy executed many thousand years ago by one or more authorities.



Left: figure 2.1.1: Globular envelope with a cluster of accounting tokens. Clay, Uruk period. From the Tell of the Acropolis in Susa. Source: Musée du Louvre, Département des Antiquités Orientales, Paris.
Middle: figure 2.1.2. A string of accounting tokens secured by a bulla of clay carrying markings proving its authenticity and integrity. Source: MS 4523, The Schøyen Collection
Right: figure 2.1.3: A stamp seal used to imprint clay as proof of identity and ownership. North Syria/Iraq/Iran, 5th-4th millennium BC. Source: MS 2411/1, The Schøyen Collection (http://www.schoyencollection.com)

(b) Selected symbols from various seals:
(1) Eridu? (2) Larsa (3) 'snake' (4) 'bird' (5) Ur
(6) Der? (7) Keš. (After Legrain 1936)

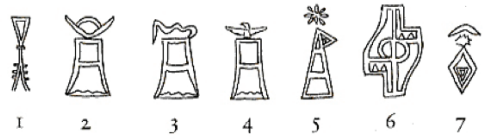


Fig 2.1.4. Example of symbols used in seals that implied authority when used in the context of good transactions across multiple cities. (Source: J.N. Postgate).

From the previous we may carefully conclude that communities, even before history was written, relied on systems supporting policy based authorization transactions. One can imagine that community legislation and agreements established between parties govern the transaction based exchange of ownership. Hereto, authorities establish and execute policies based on such legislation and agreements. Authorities arrange the execution of transactions as it must be considered fair and acceptable to the involved parties without any reason for doubt. In essence, the execution of an authorization transaction involves the evaluation and effectuation of policies

by authorities that are based on agreements embedded in the established community legislation. The capability of parties to handle an authorization transaction in such way, arranging the (temporary) transfer of ownership of the underlying subject(s), is in our context the essential function of an authorization system.

An important other lesson to be learned from the pre-historic example is that a simple to recognize system using these tokens, containers and seals proved to an effective way to arrange authorized contribution and re-distribution of wealth within communities in a trusted way. All intricacies and complexities are pre-arranged by rules that operate the system and get translated into policies that take decisions during the authorization of a transaction. The containment and sealing by a token authority represents the outcome of such decision(s) in a way that is known to be correct by all participants in the transaction. This knowledge as such, creates trust in the correctness of a decision. The outcome can be used in the future, without the need to re-asses all policy decisions underlying the authorization decision(s). Authenticity and integrity of the "message" (the sealed container/ string) containing the token(s) is an essential element to secure the decision such that it remains valid and trusted over time. Authenticity is implemented by using unique seal(s) imprinted by one or more participating authorities. Message integrity is ensured by the clay envelope / secured string containing the token(s). The policies used to create / interpret / destroy a message containing tokens provide the understanding of the meaning of a token. Note that without such knowledge, the token itself is meaningless.

The beauty provided by the simplicity of token-based implementations of authorization systems operating in the context of complex rules and policies established by collaborating groups of independent parties fascinated me such that it became a major subject of research for this thesis.

2.1.2 Origin of AAA

In the early days of the Internet, subscribers used a modem to dial via telephone lines an access point of an Internet Service Provider (ISP). After the modem connection was established, a Network Access Server (NAS) started a protocol requesting a username & password before allowing access to the Internet. To verify the correctness of the username/password combination, the NAS contacted a server containing such details for all of its subscribers. This type of server was called a "Authentication, Authorization and Accounting" server, or short "AAA Server". A protocol named "Remote Authentication Dial In User Service" (RADIUS [R2865]) was used to exchange user information between the NAS and AAA server (see fig. 2.1.5). Merit [MERIT] was responsible for operating NSFnet [NSFN], a precursor to the current Internet. The need to perform authentication, allowing dial-in users access to NSFnet, was first specified as an RFI by Merit in 1991. Livingston Enterprises [LIV] was awarded the contract and delivered the first NAS and AAA server.

By allowing AAA servers to act as a representative (proxy) for other AAA servers [R2607], an access request could be routed from a "foreign ISP" AAA server to the AAA server of the subscribers

"home ISP". This mechanism authorized users to access the Internet via different ISP's without having to subscribe to each ISP individually. Roaming agreements between ISP's arranged such type of collaboration. At business level, such roaming agreements must be in place before the policies, governing the proxy functionality of an ISP's AAA server, can be configured. Merit played a key role in developing roaming.

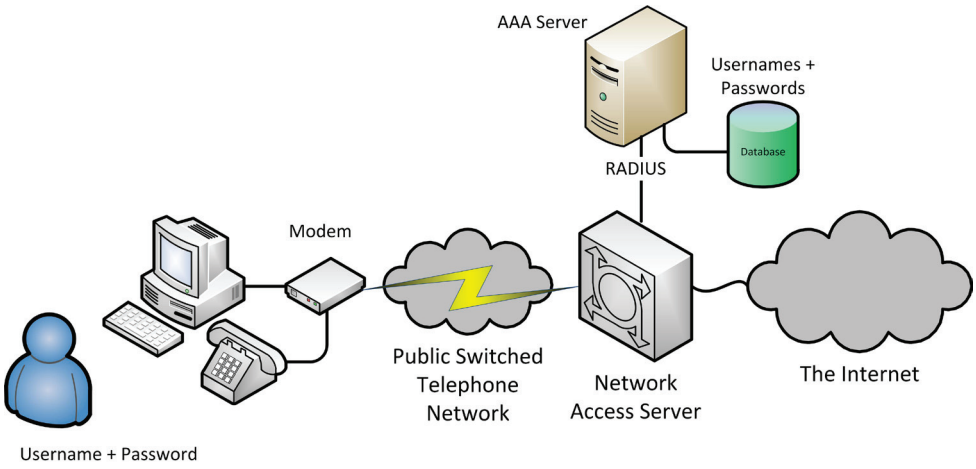


Fig 2.1.5. The role of an AAA server, allowing dial-in users to get access to the Internet.

2.1.3 Authorization in GRID context

When early Grid communities started to share compute resources, a simple mechanism that used a local "grid map file", mapping community user names to local "technical" login-in names, provided grid node access. This certificate based mechanism also allowed a grid node to proxy a community user on other nodes [GLOB]. Foster, et. al, [FOST], seeking a more scalable and granular way to arrange node access, described the concept of individuals and/or organisation forming a Virtual Organisation. A Virtual Organisation was defined by its "Sharing Rules". Sharing Rules clearly define what is shared, who is allowed to share and the condition under which sharing occurs. The Virtual Organisation Membership Service (VOMS [ALFI]) implemented this concept by means of a centrally managed repository containing more granular credentials using attribute certificates to support its communication [CIAS]. Foster's sharing rules concept represents an important requirement for service providers. Together, service providers must implement the common sharing rules in such a way that they are trusted by other service providers to offer the correct service to service requestors. The European Middleware Initiative [EMI] is an example that standardises such approaches. Initiatives [EMIS] work on the transfer of security attributes that state VO & role memberships and allowing site central administration and enforcement of common attribute-based authorization policies across different Grid compute areas.

2.1.4 Observing the Internet Engineering Task Force context

The IETF [IETF] is *an open community based effort to create technical documents that influence the way people design, use, and manage the Internet*.

We came to the IETF to look for technologies that can contribute to a solution when considering our multi-kingdom problem. Several authorization mechanisms were in place within the Internet around 1998 that were engineered by the IETF. Technologies arranged for example access to the Internet via dial-in (section 2.1.2), WiFi, mobile devices, etc. Here, authorization transactions are sequences of request and reply messages that are exchanged between entities such as a user, a network resource, an authority, the user's home ISP, etc. It arranges access based on established user rights, which are typically evaluated by policies administered by the provider(s) of a service. Typically a user will need to authenticate before authorization to access a requested service can take place. Entities, taking part in authorization transactions, may belong to the same or to different organizations. Policies are used to perform authorization decisions. At that time, the IETF felt there was a need for a common scheme, addressing various functional architectures for its authorization services [R2904I]. Based on attending IETF Working Group meetings and by studying their chartered work, some key observations could be made. These observations made us curious about answers to a number of questions that could be raised considering our multi-kingdom problem. Table 2.1.1 shows a number of such questions that were raised when observing the IETF context.

IETF observation	Question
Users accessing a single service can belong to different home organisations as for example in the RADIUS roaming case.	How multiple services, provided by different organisations, can be authorized if combined for users from different home organisations?
The IETF worked on the concept of policies within the Policy Framework Working Group [PFWG]. This group was however considering single domain cases only.	Can the policy framework span multiple domains, if each domain has a requirement to maintain autonomy?
The Policy Framework Working Group was chartered to collaborate with the Resource Allocation Protocol Working Group [RAPWG] that worked on ways to allow policy information to be exchanged between a concept called a Policy Decision Point" and "Policy Enforcement Point". These points interacted via the "Common Open Policy Server" protocol. This work complemented the ReSerVation Protocol [R2205] as a way to provide policy based admission control to resources across a network [R2750]. The RAP work mainly considered handling of policies within a single domain, where the RSVP protocol could trigger such decisions in each domain along a path separately.	Is the RAP/RSVP mechanism capable of making coordinated policy decisions across domains?

IETF observation	Question
The IETF worked on the RADIUS protocol, capable of supporting AAA functions [R2865]. At this time the IETF was considering a next generation AAA protocol capable of handling new requirements. Short-term requirements were posed by Working Groups, addressing items such as Network Access Servers [NASWG] and Mobile IP [MIPWG]. Inspired by the RADIUS roaming concept [R2607], we were thinking about constructing a network of autonomous AAA servers, each able to take part in a decision that provides an end-to-end service. Here a service request could lead to one or more requests that are forwarded to other domains in order to complete the authorization decision. In each domain, policies take decisions that could lead to additional actions and/or replies. The combination of multiple replies is subsequently needed to determine success or failure of the original request.	Is the "RADIUS" approach (and its intended new version) supporting networks of autonomous AAA servers taking policy decisions within each domain workable?
Various protocols and message objects are being used by transactions between entities taking part in authorization decisions using protocols such as RADIUS [R2865] (its intended successor DIAMETER [R3588]) and COPS [R2748], etc.	What protocols and objects (both attributes and policies) would be needed and suitable for communication of request / reply messages that are pushed / pulled across domains?
Trust must pre-exist between authorizing entities before authorization transactions can be handled. It was unclear how to arrange this. Moreover, the question of what trust really means was giving different and confusing answers.	Is trust something that should be handled as a technical issue by some protocol or key management system or are business issues also important considerations when implementing trust?

Table 2.1.1: Observations and resulting questions regarding the applicability for our multi-kingdom problem.

The observations and resulting questions of table 2.1.1 formed the foundation of the presented research in this chapter. This research lead to the description of a Generic AAA framework and architecture, capable of describing and handling authorization sequences in a distributed way as will be presented in section 2.2 and 2.3.

Note that within the IETF we started to call our multi-kingdom problem the "multi-domain authorization" problem. In our study it typically means allowing access to (pre-arranged) network and/or networked resources (e.g. computing, visualisation, experiments, etc.) that are owned by two or more domains based on some form of service agreement.

2.1.5 Related Internet Engineering Task Force work

To motivate the position of our IETF/IRTF research work, guided by the questions of table 2.1.1, we now describe the broader context around authorization and the related concept of authentication, based on work we saw evolve at the start of our research. This will explain why we could not find answers to a number of questions we had.

2.1.5.1 Authentication and Authorization related work

At the start of our research in 1998, the IETF work related to authentication and authorization can be subdivided by placing this work in into three categories:

1: Managing the security of connections between endpoints. The confidentiality, integrity of messages during data transmission can be protected, whilst ensuring the authenticity of endpoints. The authenticity of endpoints can act as a way to create a relationship between two parties based on security. Examples of such work took place at:

Level	Involved IETF work.
IP level	where cryptographic systems (supporting combinations of authentication, integrity, access control and confidentiality) were being engineered by the IP Security working group [IPSWG]
Transport level	where cryptographic systems were engineered to support confidentiality, authentication and integrity byathe Transport Level Security working group [TLSWG] supporting protocols such as TLS [R2246] and HTTPS [R2818].
Application level	where work around the secure shell [SSHWG] provided support for secure remote login, command execution, file transfer and more by encryption, authentication and compression. Next to username/password methods, a popular (mandatory) authentication method used is based on secure exchanges of public key material [X509].

Table 2.1.2: Technologies securing connections between endpoints.

Although these technologies implicitly can be considered as a way to authorize access to endpoint entities once a secure connection is established, these technologies are not designed to be aware of the underling network technology and as such provide ways to control its behaviour and quality aspects. These technologies could however be part of ensuring secure communication between entities involved in authorization sequences.

2: Communication of message objects. Managing access rights to a specific service using by means of (secured) communication of message objects, possibly involving different autonomous domains. Securing means encapsulating objects in a certain message format, such that at least message authenticity and integrity can be ensured. Some protocols did not implement such security and focussed on standardising attributes and its values.

Message object concept	Involved IETF work
Certificates	Trusting the authenticity of end-entities (users, web applications, etc) and attributes describing entity rights using a certificate based key management system. Such systems are typically based on ITU-T recommendation X.509 [X509] describing the Public Key Infrastructure (PKI) that handles public key- and attribute certificates. This work was performed by the PKIX [PKIXWG] working group
Security Tokens	Common Authentication Technologies [CATWG] with a focus on the Generic Security Service (GSS) API [R2078] allowing secret key technologies (e.g. Kerberos [R1510]) oar public key approaches (e.g. X.509) to be used as security mechanism allowing client-server applications to exchange tokens that establishes a "security context".
Protocol exchange of attributes	Communication of user and associated attributes, describing user access rights to network access devices using protocol such as RADIUS [R2865], DIAMETER [R3588] and Extensible Authentication Protocol (EAP) [R2284].
Roaming	Allowing the exchange of attributes that allows users to roam between Internet Service providers as worked on by the ROAMOPS Working Group [ROWG] defining mechanisms such as proxy chaining [R2607].

Table 2.1.3: Technologies can be used to allow an authority to assert attributes that can for example describe user authorizations or attributes that can be used in policy decisions.

3: Managing network services by policy based authorization mechanisms.
A number of groups focussed on protocols mechanisms enabling the signalling of message objects that can be used by policy based systems that determine aspects such as setting up and allocating the forwarding behaviour of network routers to packet flows.

Authorization mechanism	Involved IETF work
RSVP	The handling of QoS flows by using the Resource reSerVation setup Protocol RSVP [R2205] as setup mechanism. RSVP is designed to control a set of network devices to allow it to handle IP flows with a certain QoS in an end-to-end fashion. RSVP is a mechanism: That supports the Integrated Service Architecture [R1633] that was used for example to arrange ATM virtual circuits [MCDY]. Work on these subjects was performed by the Integrated Service over Specific Link Layers (ISSLL) working group [ISWG]. That was being engineered to configure a chain of routers with a "Per Hop Behaviour" in a network supporting Diffserv [R2998], and That was also engineered to configure an explicit QoS path using Label Switch Routers using the MultiProtocol Label Switch (MPLS) protocol [R3031]. Where extensions to its protocol for "traffic engineering" (RSVP-TE) got engineered for a generalized version of MPLS [R3473]. This mechanism allowed different types of network connections to be provisioned by a separate control plane, managing connections of, for example, optical (lambda) networks.

Authorization mechanism	Involved IETF work
Bandwidth Brokerage	Using RSVP, the (ISSLL) working group [ISWG] was engineering a Subnet Bandwidth Manager enabling LAN based access control for Layer-2 IEEE 802.1p [8021] compliant switched networks
RAP	Where the RSVP Admission Policy (RAP) [RAPWG] working group was exploring and extending the use of its Policy_Data object to establish a scalable policy control model. It used the Common Open Policy Server Protocol (COPS) [R2748] that was defined between a Policy Decision Point (PDP) and a Policy Enforcement Point (PEP). These entities are defined in a framework for policy based admission control [R2753].

Table 2.1.4 Technologies supporting the communication of policy based decisions.

As we saw it then, many of the above initiatives were using each other's results. An example of this would be: Establishing a secure TLS connection between domain bandwidth controllers as a way to ensure its authenticity by means of a shared secret or X.509 public key certificate [X509].

Given the type of problem we were considering (multi-domain authorization of network resources) we decided that the work around handling QoS flows would be a good starting point. In particular, we explored the idea to extend the PDP and PEP concepts [R2753] by placing these concepts in a multi-domain scenario. We envisaged multiple PDP's communicating as a network of PDP's, each taking policy based decisions in a distributed way, similar to a network of RADIUS AAA servers [R2865]. At this point we decided to assume that functionalities for establishing connection security (category 1) and communicating user rights / credentials (category 2) could be re-used from work performed in these areas. Extending category 3 work lead us to the Generic AAA framework and architecture considering the handling of different types of authorization sequences. We assumed to be able to re-use methods that securely communicate transactions and ensures authenticity of attributes. In this way the security (confidentiality, integrity and authenticity) of the communication between entities can be trusted. However, trusting the operation of an authorization system that is based on taking the correct policy decisions is an entirely different question that needs to be addressed.

2.1.5.2 Related initiatives in the area of policy based management

As explained, our initial approach was to study policy-based decision taking mechanisms when providing a particular QoS to an end-to-end network service. In the area of policy-based decision taking, we found that work was performed by the Directory Enabled Networking (DEN) [STRA] initiative. This initiative provided a management paradigm for network devices such as routers and switches, based on their logical role. In DEN, the behaviour of network equipment could be provisioned from a central policy repository, enabling management from a single point. DEN defined an information model that described network device management objects. It extended the Common Information Model (CIM), developed by the Desktop Management Task Force

(DMTF) [CIM]. The management objects determine the behaviour and functionality of network devices including its QoS behaviour. By centralizing its management, DEN addressed the management of a network and its services provided as a whole. The idea was born out the recognition of Cisco and Microsoft that a network could be managed out of a central directory [BERN]. The IETF provided a Lightweight Directory Access Protocol (LDAP) [R2251] as a way to access such a directory. The DEN/CIM way of thinking was brought to the IETF by the Policy Framework Working Group. As explained, it was chartered to only consider single-domain solutions [PFWG]. Based on this observation, we wanted to consider an authorization architecture that is agnostic to where policies and its attributes are stored and how they are communicated.

2.1.6 Related Research

Research on security concepts for data processing environments became an important topic in the 1970's when computers became multi-user, multi-application systems allowing remote access. Another important driver for research in the area of security was based on governments realizing its increasing dependency on computer technology. Based on such concerns, the US government's Office of Management and Budget [CLIN] put for example the National Institute of Standards and Technology [NIST] in charge of providing security guidance [OMB] to heads of departments and agencies. Research on mechanism arranging the security of information processing formed the base of various Federal Information Processing Standards [FIPS] and other security standards such as the ISO/IEC 27000 series standards [ISO27] and ANSI [INCI].

In the area of authorization, significant research effort was put into formalizing the Role Based Access Control (RBAC) model [FERR]. Since the 1990's the RBAC model was gaining popularity as generic access control model solving inherent management scalability issues of the traditional Mandatory Access Controls (MAC) and Discretionary Access Controls (DAC) [SAN96, SAN98, OSBO] models. RBAC was a model allowing central rather than distributed administration of access policies. The RBAC model assumes that all activities in a system are conducted through transactions, rather than having individuals manage access privileges to objects under their control (DAC) or by classifying objects requiring a clearance level (MAC). Both DAC and MAC mechanisms are defined in the Trusted Computer System Evaluation Criteria [TCSE].

The RBAC model formally describes three basic rules that are required to execute a transaction by a user (table 2.1.5). RBAC ensures that all access is provided through roles. A role is a collection of permissions. Users only get permissions through the roles to which they are assigned to or they inherit from a hierarchy of roles. Users are considered members of roles. Roles can be defined independently of permissions. As such, RBAC simplified the way permissions are managed in organizations. Roles in an organization do not tend to change as frequently as a person's assignment to a role.

No.	Rule	Description
1	Role assignment	All active users are required to have some active role
2	Role authorization	A subject's active role must be authorized for the subject. This rule ensures that users can take on only roles for which they are authorized
3	Transaction authorization	A subject can execute a transaction only if the transaction is authorized for the subject's active role. With (1) and (2), this rule ensures that users can execute only transactions for which they are authorized

Table 2.1.5: RBAC basic rules.

RBAC works very well in environments where access control to information objects must be arranged. The research on RBAC indicated that the model is not the panacea of all access control issues [FERR], in particular where sequences of operations need to be controlled. Around 2000 efforts were made to propose RBAC as a standard [SAN00], which got adopted in 2004 by ANSI/INCITS as standard 359-2004 [INCI].

Subsequent research was put into the use of the RBAC model with mechanisms that can explicitly specify, allocate and communicate privileges of users by means of X.509 Attribute Certificates (AC's). Such "role assignment" certificates issued by a "privilege allocator" can be stored in a central (LDAP) directory for subsequent use by "privilege verification subsystems". The PERMIS project from University of Kent (UK) researched this concept [CHA1]. It effectively created a "Privilege Management Infrastructure (PMI)" using Attribute Certificates binding a user's name to one or more privilege attributes securely issued by a trusted Attribute Authority. A similar approach to perform distributed management of access rights was researched by the AKENTI project from Lawrence Berkley National Laboratory (USA) [JOHN]. A comparison is provided by Dave Chadwick [CHA2]. Efforts were also put into the research of policy languages arranging authorization such as Ponder [DAMI]. Such efforts evolved with work performed in the Grid/HPC context in the Open Grid Service Architecture Authorization (OGSA-Authz) Working Group [OGSA]. Similar requirements that evolve DAC into RBAC and AC's now appear in the context of the Big Data processing [RAMA]. Hadoop HDFS [SHVA] is a DAC based file-system. Efforts are on-going in authorizing access to data elements stored in Hadoop using formats such as Parquet [PARQ].

At UvA, related research in the area of authorization was performed in particular by Yuri Demchenko on arranging granular access control to laboratory equipment [DEM1,DEM2]. Considering Grid scenario's, he also researched the use of XML based languages such as SAML and XACML to express and communicate dynamic security context information in policy based access control mechanisms [DEM3].

2.2 AAA Authorization Framework[1]

The IRTF AAA Architecture Research Group [AAAARG] performed research into fundamental authorization sequences and described the outcome as framework models in RFC2904 "AAA Authorization Framework" [R2904] and RFC2903 "Generic AAA Architecture" [R2903] described in section 2.3. Further study into the sequence models continued within the OGF AuthZ Working Group [OGSA] and were refined in GFD-I.38 [GFD38]. Both studies showed that there are at least three fundamental sequences that describe the interaction between basic entities involved in an authorization. In this section we first (in summary) explore the entities and sequence models within networking and grid environments. An another important sequence, called the "token sequence" was added over time. Sections 2.2.7 until 2.2.14 are added for completeness in their original form. These sections represent the implementation idea's we had at that time considering session management, storing authorization information using Attribute Certificates, resource management, message forwarding and delivery, end-to-end security and process handling.

2.2.1 Authorization Entities and Trust Relationships

The AAA Authorization Framework recognizes the basic conceptual entities shown in fig 2.2.1.

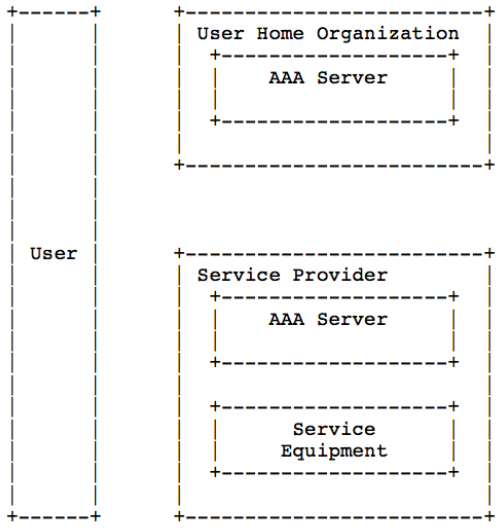


Fig. 2.2.1 -- The Basic Authorization Entities

1 This section is based on: RFC2904 "AAA Authorization Framework" J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, IETF August 2000 and GFD.38 "Conceptual Grid Authorization Framework and Classification", M. Lorch, B. Cowles, R. Baker, L. Gommans, P. Madsen, A. McNab, L. Ramakrishnan, K. Sankar, D. Skow, M. Thompson, Open Grid Forum, Nov. 2004.

The entities may be participants in an authorization scenario, and can be defines as:

1) A **User** who wants access to a service or resource.

2) A **User Home Organization** (UHO) that has an agreement with the user and checks whether the user is allowed to obtain the requested service or resource. This entity may carry information required to authorize the User, which might not be known to the Service Provider (such as a credit limit).

3) A Service Provider's **AAA Server** which authorizes a service based on an agreement with the User Home Organization without specific knowledge about the individual User. This agreement may contain elements that are not relevant to an individual user (e.g., the total agreed bandwidth between the User Home Organization and the Service Provider).
4) A Service Provider's **Service Equipment**, which provides the service itself. This might, for example, be a NAS dial-in service, or a QoS network routing service, or a print server in the Internet Printing service.

During subsequent research in the OGF AuthZ WG described in GFD.38 [GFD38], the RFC2904 concepts were more generalized and described as:

1) **The User or Subject:** An entity (e. g. a person or process) that can request, receive, own, transfer, present or delegate an electronic authorization as to exercise a certain right.

2) **The User Home Organization:** The Organization that administers a user by determining and providing attributes that describe a User (e.g. access rights, quota, roles, etc.) that may be evaluated during a policy decision.

3) **Authorization Authority or AAA Server:** An administrative entity that is capable of and authoritative for issuing, validating and revoking an electronic means of proof such that the named subject (a.k.a. holder) of the issued electronic means is authorized to exercise a certain right or assert a certain attribute. Right(s) may be implicitly or explicitly present in the electronic proof. A set of policies may determine how authorizations are issued, verified, etc. based on the contractual relationships the Authority has established. Specifically (see fig. 1) the AAA server is considered the single authority governing access to the underlying service equipment. The AAA Server and Service Equipment form in this sense a unity called the "service provider". An AAA server can also represent a User Home Organization containing user access rights.

4) **The Service Equipment or Resource.** The entity that represents the service, which needs information that authorizes the usage of the service offered by the equipment. A component of the system that provides or hosts services and may

enforce access to these services based on a set of rules and policies defined by entities that are authoritative for the particular resource.

When considering the above, it is important to recognize that the term Authorization can mean multiple things. GFD.38 therefore recognizes that the term **authorization** can mean:

1) the process of issuing a proof of right
2) the proof of right (or reference to) itself (i.e., an authorization token)
3) the process of making an authorization decision by checking a proof of right, e.g., by rendering user attributes against access control policies

It is also important to understand that when studying authorization, it is assumed that the parties who are participating in the authorization process have already gone through an authentication phase. Although many systems combine Authentication, Authorization and sometimes Accounting functions, verifying and managing the identities of a user is considered a separate topic and is therefore not considered in detail in this thesis.

2.2.2 Authorization message sequences

In summary, between the authorization entities that RFC2904 describes, messages are exchanged to handle an authorization transaction. RFC2904 distinguishes different message sequences that request authorization and subsequently use the authorization to gain access to a resource. RFC2904 first considers the single domain case where the service provider itself administers the user. To request an authorization, the user can either contact the AAA Server or the Service Equipment. In the latter case, the Service Equipment will out-source the access decision to the AAA server. The AAA server will reply a decision. The Service Equipment can subsequently enforce the access based on this reply. Fig. 2.2.2 shows this sequence that is named the pull sequence. In the second and third scenario the User will send a request to the AAA server. In the second scenario, called the agent sequence (fig. 2.2.3), the AAA Server will act as an agent that will first take an authorization decision. It will then talk directly to the Service Equipment to permit or deny access. After a decision has been made in the third scenario, the AAA server will create a (secure) token that is handed back to the user. The user must then push this token to the Service Equipment hence its name: the push sequence (fig. 2.2.4).

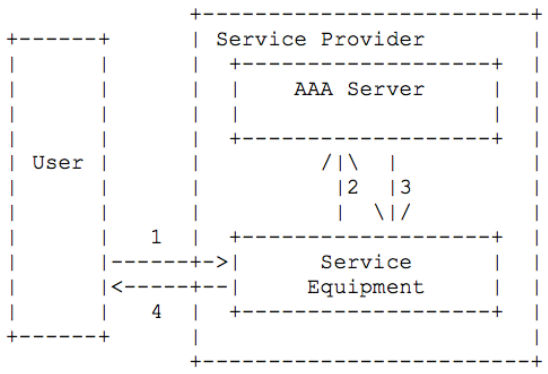


Fig. 2.2.2 -- Pull Sequence

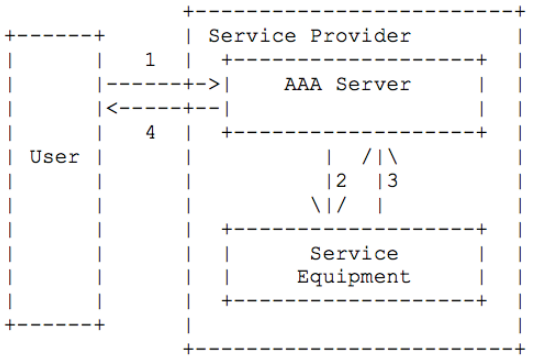


Fig. 2.2.3 -- Agent Sequence

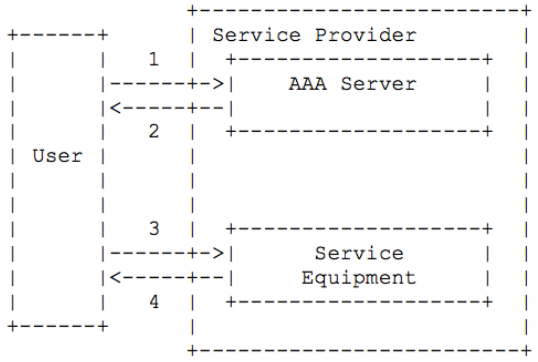


Fig. 2.2.4 -- Push Sequence

2.2.3 Roaming sequences

Cases, where the organization that authorizes (and typically also authenticates) the User is different from the organization providing the service are considered by RFC2904 as roaming

cases. In such case, the User Home Organisation administers the user. The same agent-, pull- and push sequences are possible with roaming. Fig. 2.2.5 shows an example of the roaming pull sequence. An example of roaming is where universities allow WiFi Internet access to students from other universities via a system called "eduroam" [EDUR]. In this case, the university that is visited by the foreign student is a service provider for this student acting as the User. When a student likes to access the Internet via a WiFi access point (the service equipment), the AAA server of the visited university will contact the AAA server of the students home university to verify if this particular student is registered with the home university. Note that in this case the visited university is the party that authorizes access based on information obtained from the student's user home organisation.

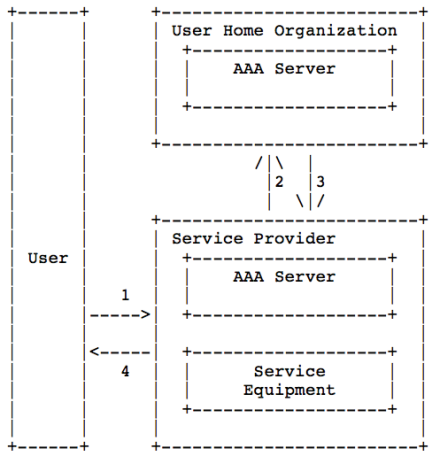


Fig. 2.2.5 -- Roaming Pull Sequence

2.2.4 Service Agreements

RFC2904 recognizes that there may be bilateral agreements between pairs of organizations involved in an authorization transaction. Agreements between organizations may take the form of formal contracts or Service Level Agreements. Fig. 2.2.6 uses double lines to show relationships that may exist between the User and the User Home Organization and between the User Home Organization and the Service Provider.

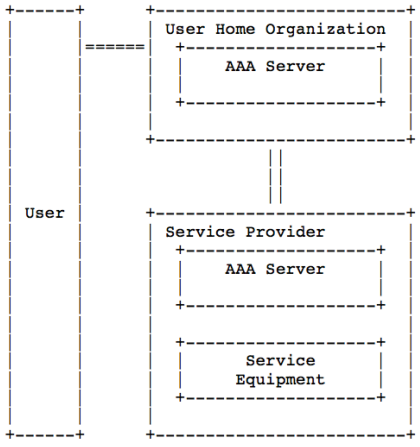


Fig. 2.2.6 -- Service Agreements

Authorization is based on these bilateral agreements between entities. The fulfilment of the User's request depends on both agreements being honoured. Note that these agreements may be implemented by hand configuration or by evaluation of Policy data stored in a Policy database. Note that arranging bilateral agreements becomes a scalability issue when the amount of participants and relationship complexity increases.
A very import point is that *there must be a set of known rules in place* between entities in order to execute authorization transactions. Trust is necessary to allow each entity to "know" that the policy it is authorizing is correct. This is a business issue as well as a protocol issue. As said in the introduction of this chapter, this recognition will be further elaborated in chapter 5.

In RFC2904 we stated: *Trust is often established through third party authentication servers (such as Kerberos), via a certificate authority, by configuring shared secrets or passwords, or by sharing a common facility (such as a connecting wire between processors). These "static" trust relationships are necessary for authorization transactions to take place. Static trust relationships are used in an authorization sequence to establish a "dynamic" relationship between the User and the Service Equipment.*

Although we said that "trust relationships are necessary for authorization transactions to take place" we will see in chapter 5, that it will take more than the actions mentioned here to establish these trust relationships.

2.2.5 Distributed (multi-domain) services

RFC2904 considers services that are provided by more than one Service Provider acting in concert is a distributed service. RFC2903 (See section 2.3) introduces how requests can be handled in such cases. In chapter 5 we will recognize that acting in concert will mean that the collaborative group must have standards and rules that each member translates into conforming policies such that each Service Provider is able to correctly contribute to the service.

Figure 2.2.7 illustrates distributed services. The double lines in fig. 2.2.7 represent some form agreements that must pre-exist before authorization sequences between the participants can take place.

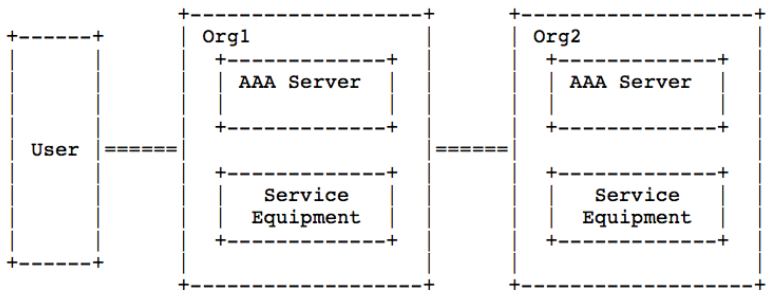


Fig. 2.2.7 -- Distributed Services

These agreements may arrange for example that only Org 1 has awareness of the User and that Org1 will act as a proxy for Org2. Such agreements may be established in a bi-lateral way. Note that when the amount of Service Provider Organisations increases, establishing bi-lateral agreements do not scale well and it may be increasingly difficult to create end-to-end service that is always uniform amongst arbitrary combinations of organisations. Such recognition has been the basis to the study of chapter 5 into what is needed to establish a Service Provider Group that arranges the "acting in concert" in such a way that it is uniform and trusted by all participants.

Using the previous concepts one can describe possible sequences between User, Org1 and Org2 where the sequence between the User and Org1 can be different of the sequence between Org1 and Org2. The User and Org1 might use a pull sequence, and the second might use an agent sequence, where Org1 acts as the "User" of Org2. This example is illustrated in figure 2.2.8.

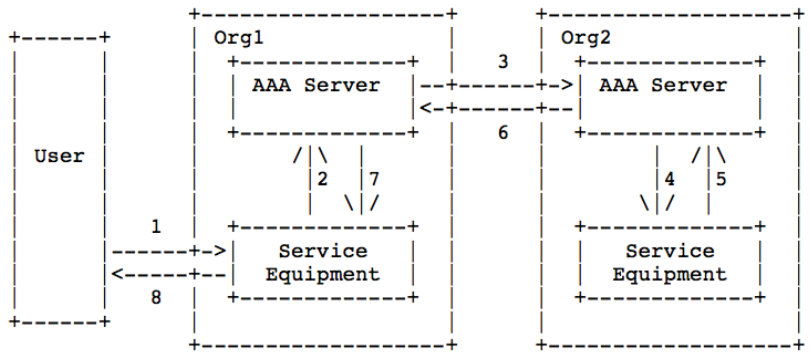


Fig. 2.2.8 -- A Possible Distributed Sequence

When combined with roaming as shown in fig. 2.2.9, one can imagine several contract and trust relationships that may be set up in number of ways, depending on a variety of factors, especially the business model. New entities that combine or add capabilities can be created to meet business needs.

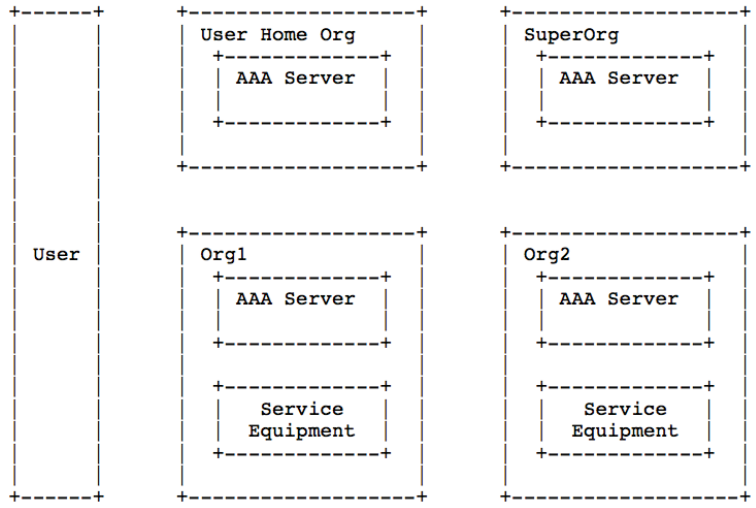


Fig. 2.2.9 -- Roaming and Distributed Services

In fig. 2.2.9, one such possibility, a SuperOrg entity is shown. The idea is that this entity would provide authentication and authorization for organizations that are providing services to end-users. It could be considered to be a wholesaler or broker. While not all authorization will require having a broker, authorization protocols should allow such entities to be created to meet legitimate requirements. In this sense this example shows that the RFC2904 AAA Authorization Framework offers a way to discuss how authorization transactions sequences can be designed and discuss their requirements.

2.2.6 Hybrid sequences

In our further research (see section 3.4.2) we recognized that the pull-, push-, and agent sequence are elementary sequences that can be combined to create solutions. Fig. 2.2.10 shows an example of a combined agent- and push sequence. Here a User requests a resource from the AAA server (1). After an authorization decision has been taken, the AAA provisions the Service Equipment (2/3) with information (e.g. service parameters and key information). Based on information received from the AAA server (4) the user creates a (future) request and pushes it (5) to the service. All relevant information to honour this request is known by the Service Equipment based on it being provisioned by the AAA server (2). The Service Equipment might provide the AAA server with additional service details (3): e.g. which service port to use. A token, which points to the agreed resource, is handed in a secured way (e.g. using a secure hashing algorithm) to the User (4). At a later point in time the User then pushes this token to the Service Equipment (5). The provisioned information (2) allows a token to be recognized as being authentic. The token also points to the service to be provisioned. The service typically then acknowledges the request (6) as a sign that the User is able to use the requested service.

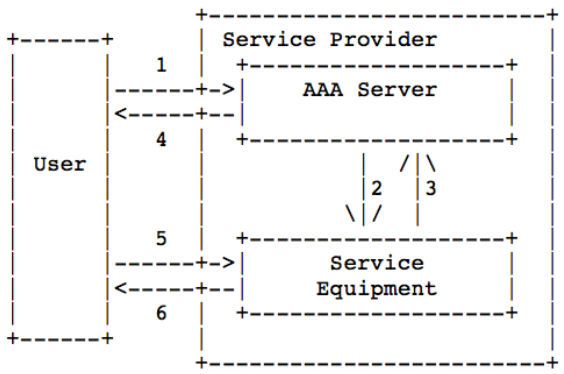


Fig. 2.2.10 – A hybrid agent/push sequence

2.2.7 Relationship of Authorization and Policy

As stated in section 2.2, sections 2.2.7 - 2.2.14 are included for completeness only.

The Policy Framework (policy) Working Group [POLWG] is seeking to provide a framework to represent, manage, and share policies and policy information in a vendor-independent, interoperable, scalable manner [R3060, PFDR, PFLDR]. This section explores the relationship of policy and authorization and sets the stage for defining protocol requirements for supporting policy when included as part of multi-domain authorization. The work presented here builds on the policy framework, extending it to support policy across multiple domains.
One view of an authorization is that it is the result of evaluating policies of each organization that has an interest in the authorization decision. In this document the assumption is that

each administration may have policies which may be indexed by user, by service, or by other attributes of the request. The policies of each administration are defined independently of other administrations.

Each independent policy must be 1) retrieved, 2) evaluated, and 3) enforced.

2.2.7.1 Policy Retrieval

Policy definitions are maintained and stored in a policy repository [PFDR] by (or on behalf of) the organization that requires them. The Policy Framework WG [POLWG] is working on a way to describe policy [PFLDR]. Other implementations describe policy as a set of ACL lists. Policy definitions must be retrieved in order to be evaluated and enforced. Policy Definitions can be indexed by requester, by service attribute, or by some other key. The organization requiring the policy is also responsible for determining which policy is to be applied to a specific authorization request.

Policy retrieval is typically done by the administration that defines the policy or by an agent acting for that administration. Thus a policy defining the times of day that a particular User is allowed to connect to the network is maintained and retrieved by the User Organization. A policy defining a time that ports will be unusable because of maintenance is maintained and retrieved by the Service Provider.

Note that some implementation may choose to have the Service Provider retrieve a policy from the User Home Organization using a distributed directory access protocol. This may be appropriate in some cases, but is not a general solution. To understand why; suppose the remote administration and the home administration communicate via a broker, which proxies their communication. In such a case the Service Provider and Home Organisation administration have no prior relationship. Therefore, the Home Organisations directory is unlikely to allow access to the remote Service Provider administration and vice versa.

2.2.7.2 Policy Evaluation

Evaluation of policy requires access to information referenced by the policy. Often the information required is not available in the administration where the policy is retrieved. For example, checking that a user is allowed to login at the current time can readily be done by the User Home Organization because the User Home Organization has access to current time. But authorizing a user requiring a 2Mb/s path with less than 4 hops requires information available at a Service Provider and not directly available to the UHO, so the UHO must either 1) have a way to query a remote administration for the needed information or 2) forward the policy to the remote administration and have the remote administration do the actual evaluation or 3) attempt somehow to "shadow" the authoritative source of the information (e.g. by having the Service Provider send updates to the UHO).

Applications might support either 1) or 2), and a general authorization protocol must allow both. Case 3) is not considered further as shadowing, involving the complexity of managing its state, seem too "expensive" to be supported by an AAA protocol.

An example of case 1 is when a Service Provider forwards a request to a UHO which includes a query for the clearance code of the User. The Service Provider policy includes reference to the clearance code and the information in the reply is used as input to that policy.

An example of case 2 is when the UHO approves an authorization conditional on the Service Provider confirming that there is currently a specific resource available for its use. The UHO includes the "policy" along with a conditional authorization to the Service Provider.

2.2.7.3 Policy Enforcement

Policy Enforcement is typically done by the Service Provider on the Service Equipment. The Service Equipment is equivalent to the Policy Target described in the Policy Framework [PFDR]. Thus a NAS may enforce destination IP address limits via "filters" and a Router may enforce QoS restrictions on incoming packets. The protocol that sends the information between the Service Equipment and the Service Provider AAA Server may be specific to the Service Equipment, but it seems that the requirements are not different in kind from what is required between other AAA servers.

In particular, an AAA Server could send a "policy" to the Service Equipment stating what the equipment should do under various situations. The Service equipment should either set up to "enforce" the policy or reject the request.

The AAA Server could also send a query to the Service Equipment for information it requires to evaluate a policy.

2.2.7.4 Distributed Policy

A policy is retrieved by a Policy Retrieval Point (PRP) from a Policy Repository, evaluated at a Policy Decision Point (PDP) or Policy Consumer, and enforced at a Policy Enforcement Point (PEP) or Policy Target [PFDR].

Generally, any of the AAA Servers involved in an authorization transaction may retrieve a policy or evaluate a policy, and any of the Service Equipment may enforce a policy. Policy Repositories may reside on any of the AAA Servers or be located elsewhere in the network.

Information against which policy conditions are evaluated (such as resource status, session state, or time of day) are accessible at Policy Information Points (PIPs) and might be accessed using

Policy Information Blocks (PIBs). An interesting question in any authorization application that uses policy is where are the PDPs, PRPs, PIPs and PEPs?

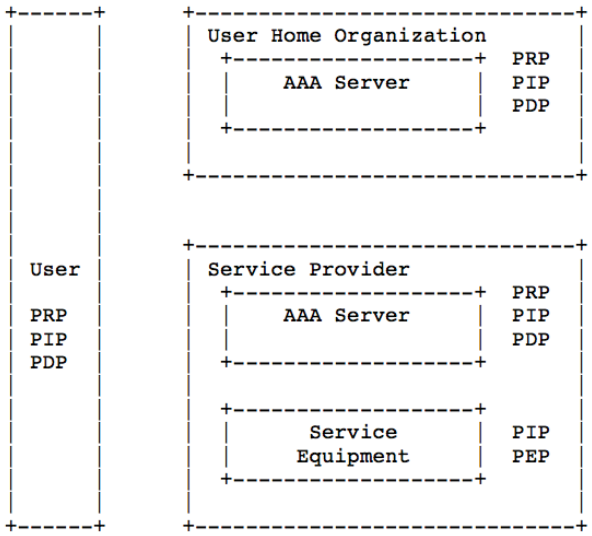


Fig. 2.2.11 -- Where Different Policy Elements May be Located

Figure 2.2.11 shows which policy elements may be available at different points in the model. In distributed services, there may be multiple Service Providers involved in the authorization transaction, and each may act as the policy elements shown below. Note that the User (or requester) may also be a PRP (e.g. use policy description to specify what service is being requested), a PIP (have information needed by other entities to evaluate their policy), and a PDP (decide if it will accept a service with specific parameters).

An AAA protocol must be able to transport both policy definitions and the information needed to evaluate policies. It must also support queries for policy information.

2.2.8 Use of Attribute Certificates to Store Authorization Data

This section outlines another mechanism that could be used for securely transporting the attributes on which an authorization decision is to be made. Work on X.509 Attribute Certificates is currently being undertaken in the Public Key Infrastructure (PKIX) Working Group [PKIX]. This proposal is largely based on that work.

When considering authorization using certificate-based mechanisms, one is often less interested in the identity of the entity than in some other attributes, (e.g. roles, account limits etc.), which should be used to make an authorization decision.

In many such cases, it is better to separate this information from the identity for management, security, interoperability or other reasons. However, this authorization information may also need to be protected in a fashion similar to a public key certificate. The name used here for such a structure is an Attribute Certificate (AC) which is a digitally signed (certified) set of attributes. An AC is a structure that is similar to an X.509 public key certificate [R2459] with the main difference being that it contains no public key. The AC typically contains group membership, role, clearance and other access control information associated with the AC owner. A syntax for ACs is also defined in the X.509 standard.

When making an access decision based on an AC, an access decision function (in a PEP, PDP or elsewhere) may need to ensure that the appropriate AC owner is the entity that has requested access. The linkage between the request and the AC can be achieved if the AC has a "pointer" to a Public Key Certificate (PKC) for the requester and that the PKC has been used to authenticate the request. Other forms of linkage can be defined which work with other authentication schemes.

As there is often confusion about the difference between public key certificates (PKC's) and attribute certificates (ACs), an analogy may help. A PKC can be considered to be like a passport: it identifies the owner, it tends to be valid for a long period, it is difficult to forge, and it has a strong authentication process to establish the owner's identity. An AC is more like an entry visa in that it is typically issued by a different authority than the passport issuing authority, and it doesn't have as long a validity period as a passport. Acquiring an entry visa typically requires presenting a passport that authenticates that owner's identity. As a consequence, acquiring the entry visa becomes a simpler procedure. The entry visa will refer to the passport as a part of how that visa specifies the terms under which the passport owner is authorized to enter a country.

In conjunction with authentication services, ACs provide a means to transport authorization information securely to applications. However, there are a number of possible communication paths that an AC may take.

In some environments, it is suitable for a client to "push" an AC to a server. This means that no new connections between the client and server domains are required. It also means that no search burden is imposed on servers, which improves performance.

In other cases, it is more suitable for a client simply to authenticate to the server and for the server to request the client's AC from an AC issuer or a repository. A major benefit of the this model is that it can be implemented without changes to the client and client/server protocol. It is also more suitable for some inter-domain cases where the client's rights should be assigned within the server's domain, rather than within the client's "home" domain.

There are a number of possible exchanges that can occur, and there are three entities involved: client, server, and AC issuer. In addition the use of a directory service as a repository for AC retrieval may be supported.

Figure 2.2.12 shows an abstract view of the exchanges that may involve ACs. Note that the lines in the diagram represent protocols which must be defined, not data flows. The PKIX working group will define the required acquisition protocols. One candidate for the lookup protocols is LDAP (once an LDAP schema exists which states where an AC is to be found).

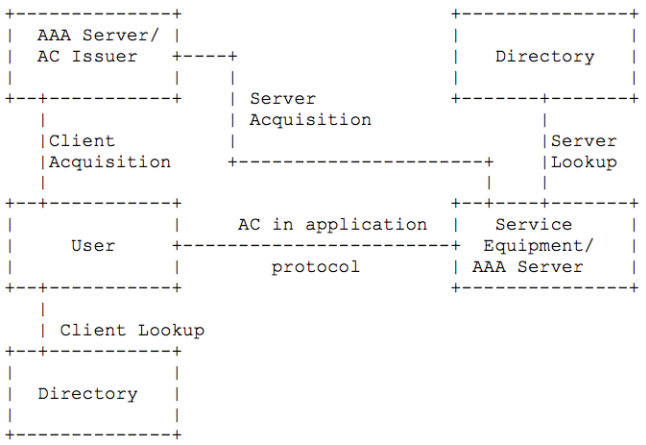


Fig. 2.2.12 -- AC Exchanges

Figure 2.2.13 shows the data flows which may occur in one particular case, that termed "push" above (section 2.2.2).

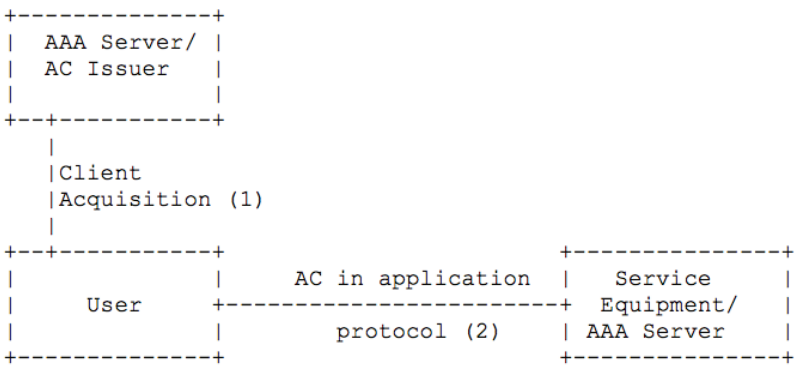


Fig. 2.2.13 -- One example of an AC exchange

In the diagram, the user first contacts the AC Issuer and then incorporates the AC into the application protocol. The Service Equipment must then validate the AC and use it as the basis for the access decision (this functionality may be distributed between a PEP and PDP).

2.2.9 Resource Management

Authorization requests may be chained through a set of servers, as described in previous sections. Each of the servers may have a contractual relationship with servers on either side of it in the chain. In many of the applications being considered, the authorization results in establishing of an ongoing service which we call a session. Each of the servers involved in the authorization may also want to keep track of the state of the session, and be able to effect changes to the session if required. To make it simple to discuss this capability, we assume that each AAA Server MAY have a Resource Manager component. Resource Managers tracking the same session need to be able to initiate changes to the session, and to inform other Resource Managers when changes occur. Communication between Resource Managers creates requirements for an authorization protocol.

An example of the use of resource management might be a user which sets up a QoS path through two ISPs, and while this path is active, one of the ISPs gets a request for more bandwidth from a higher priority user. The ISP may need to take some bandwidth from a the lower priority user's previously allocated session and give it to the new request. To do this, each of the administrations in the authorization path must be informed and agree to the change (this could be considered to be "authorizing the new value").

2.2.9.1 Session Management and State Synchronization

When an AAA Server grants authorization of some resource to an AAA requester (either a User or another AAA Server), the server may need to maintain session state information. This is used to make decisions about new sessions based on the state of current sessions, and to allow monitoring of sessions by all interested AAA Servers.

Each session is identified by a ***session identifier***, which ***must be unique*** within each AAA Server. Communication between AAA Servers must include the session identifier. It is desirable that the session identifier is the same across all AAA servers, otherwise each server will have to map identifiers from other servers to its own identifiers. A single session identifier significantly simplifies auditing and session control functions.

Maintaining session state across AAA administrative boundaries increases the complexity of the problem, especially if each AAA Server in the trust chain must keep state as well. This can be viewed as an inter-domain database replication problem. The protocol must include tools to help manage replicated state. Some of the problems to be addressed are:

a) Service Equipment must be able to notify its Resource Manager when a session terminates or changes state in some other way. The Resource Manager must inform other Resource Managers which keep state for this session.
b) The Resource Manager will need to set a time limit for each session which must be refreshed by having the Resource Manager query for authoritative status or by having

the authoritative source send periodic keep alive messages that are forwarded to all Resource Managers in the authorization chain. Determining the appropriate session lifetime may be application specific and depends on the acceptable level of risk. If the service being offered is billed based on time, the session lifetime may need to be relatively small; if the service is billed on usage, the lifetime may be relatively large.

c) Any Resource Manager in the chain must have the ability to terminate a session. This requires the Resource Manager to have knowledge of at least the adjacent AAA Servers in the authorization chain.

An example of how resource management can be used is in the PPP dial-in application. A home ISP may wish to restrict the number of concurrent sessions that a user can have at any given time. This is particularly important when service providers give all-you-can-eat Internet access. The possibility for fraud is quite large, since a user could provide his or her username/password to many people, causing a loss of revenue. Resource management would allow the home ISP AAA server to identify when a user is active and to reject any authorization request for the user until termination indication is received from the NAS or until the session expires.

2.2.9.2 The Resource Manager

The Resource Manager is the component which tracks the state of sessions associated with an AAA Server or Service Equipment. It also may allocate resources to a session (e.g. IP addresses) and may track use of resources allocated by peer resource managers to a session (e.g. bandwidth in a foreign administrative domain). The resource manager also provides interfaces to allow the User to acquire or release authorized sessions.

The Resource Manager maintains all session specific AAA state information required by the AAA Server. That state information may include pointers to peer Resource Managers in other administrative domains that possess additional AAA state information that refers to the same session. The Resource Manager is the anchor point in the AAA Server from which a session can be controlled, monitored, and coordinated even if that session is consuming network resources or services across multiple Service Provider administrative domains.

The Resource Manager has several important functions:

a) It allows a Service Provider operations staff to inspect the status of any of the allocated resources and services including resources that span foreign Service Provider administrative boundaries. The peer Resource Managers will cooperatively share only the state information subset that is required to assist in diagnosing cross-domain trouble tickets. The network operator may also modify or altogether cancel one of the User's active authorizations.

b) It is the process contacted by other Resource Managers to inform the AAA Server that a specific session has been cancelled. This information is relayed to the other peer

Resource Managers that also know about that session and hence must cancel it.

c) The Resource Manager conceals the identity and location of its private internal AAA components from other administrative domains and from the User, while at the same time facilitating cooperation between those domains.

d) The Resource Manager cooperates with "policy servers" or Policy Decision Points (PDPs). The Resource Manager maintains internal state information, possibly complex cross-administrative domain information, supported by dialogues with its peer Resource Managers. A policy server can use the state information when evaluating a particular policy.

e) The separation of the Resource Manager and the policy server into two distinct architectural components allows a single session to span multiple administrative domains, where each administrative domain has one or more policy server cooperating with its Resource Manager.

AAA resource managers will normally use the same trust relationships needed for authorization sequences. It is possible for independent relationships to be established, but that is discouraged.

2.2.10 AAA Message Forwarding and Delivery

An AAA Server is responsible for securely forwarding AAA messages to the correct destination system or process in the AAA infrastructure. Two well-known examples are forwarding AAA messages for a roaming AAA service, and forwarding AAA messages for a distributed AAA service. The same principle can also be applied to intra-domain communications. The message forwarding is done in one of two modes.

The first mode is when an AAA server needs to forward a message to a peer AAA server that has a known "logical destination address" that must be resolved by an application-specific procedure into its actual network address. Typically the forwarding procedure indexes into a database by an application-specific identifier to discover the peer's network address. For example, in the roaming dial-in application, the application-specific identifier may be an NAI. A bandwidth brokerage application would use other search indices unique to its problem domain to select the addressed peer AAA server. After the address resolution procedure has completed successfully, then the AAA server transmits the message to its peer over the connection associated with that destination network address.

The second mode is when the AAA server already has an established session representing an authorization. The session's state contains the addressing and context used to direct the message to its destination peer AAA server, PDP, PEP, or User. The message is sent over the AAA server's connection to that destination peer, multiplexed with other session's messages. The message must be qualified by a session identifier that is understood by both end points. The AAA message's destination may be either intra-administrative domain, or inter-administrative domain. In the former case, the destination process may reside on the same system as the AAA server.

In addition to the above message forwarding processing, the underlying message delivery service must meet the following requirements:

- Unicast capability -- An end system can send a message to any other end system with minimal latency of session setup/disconnect overhead messages, and no end system overhead of keeping state information about every potential peer.
- Data integrity and error detection -- This data transport protocol assumes an underlying datagram network layer service that includes packet discard on error detection, and data integrity protection against third party modifications.
- Reliable data transport assurance -- When an end system successfully receives a message marked receipt requested, it must acknowledge that message to the sending system by either piggybacking the acknowledgement on an application-specific reply message, or else as a standalone acknowledgement message. The sending system maintains a retry timer; when the timer expires, the sending system retransmits a copy of its original message. It gives up after a configurable number of unsuccessful retries.
- Sequenced data delivery -- If multiple messages are sent between a pair of end systems, those messages are delivered to the addressed application in the same order as they were transmitted. Duplicates are silently suppressed.
- Responsive to network congestion feedback -- When the network enters into congestion, the end systems must detect that condition, and they must back off their transmission rate until the congestion subsides. The back off and recovery algorithms must avoid oscillations.

2.2.11 End-to-End Security

When AAA servers communicate through intermediate AAA servers, such as brokers, it may be necessary that a part of the payload be secure between the originator and the target AAA server. The security requirement may consist of one or more of the following: end-to-end message integrity, confidentiality, replay protection, and nonrepudiation. Furthermore, it is a requirement that intermediate AAA servers be able to append information such as local policy to a message before forwarding the message to its intended destination. It may also be required that an intermediate AAA Server sign such appended information.

This requirement has been clearly documented in [R2607], which describes many current weaknesses of the RADIUS protocol [R2138] in roaming networks since RADIUS does not provide such functionality. One well-known attack is the ability for the intermediate nodes to modify critical accounting information, such as a session time.

Most popular security protocols (e.g. IPSec, SSL, etc.) do not provide the ability to secure a portion of the payload. Therefore, it may be necessary for the AAA protocol to implement its own security extensions to provide end-to-end security.

2.2.12 Streamlined Authorization Process

The techniques described above allow for great flexibility in distributing the components required for authentication and authorization. However, working groups such as Roamops and MobileIP have identified requirements to minimize Internet traversals in order to reduce latency. To support these requirements, data fields necessary for both authentication and authorization should be able to be carried in a single message set. This is especially important when there are intermediate servers (such as Brokers) in the AAA chain.

Furthermore, it should be possible for the Brokers to allow end-to-end (direct) authentication and authorization. This can be done as follows. The User Home Organization generates a ticket which is signed using the UHO's private key. The ticket is carried in the accounting messages. The accounting messages must flow through the Broker since the Broker is acting as the settlement agent and requires this information. There are Brokers that will require to be in the authentication and authorization path as well since they will use this information to detect fraudulent activity, so the above should be optional.

In order for end-to-end authentication and authorization to occur, it may be necessary for the Broker to act as a certificate authority. All members of the roaming consortium would be able to trust each other (to an extent) using the certificates. A Service Provider's AAA server that sends a request to the Broker should be able to receive a redirect message which would allow the two peers (Service Provider and UHO) to interact directly. The redirect message from the Broker should include the UHO's certificate, which eliminates the Service Provider from accessing the certificate archive. The request from the Service Provider could include its own certificate, and a token from the Broker's redirect message that is time stamped and guarantees that the Service Provider is in good standing with the Broker. This eliminates the home domain from accessing the Certificate Revocation List (CRL).

2.2.13 Summary of the Authorization Framework

The above has introduced the basic players in an authorization transaction as User, User Home Organization, Service Provider's AAA Server, and Service Equipment. It has discussed relationships between entities based on agreements or contracts, and on "trust". Examples of authorization sequences have been given.
Concepts of roaming and distributed services have been briefly described. Combination of roaming and distributed services was also considered and the concept of a "wholesaler" or Broker was introduced. We have considered the use of policies and attribute certificates to store and transmit authorization data. We discussed the problem of managing the resources to which access has been authorized including the problem of tracking state information for session-oriented services, and we defined the Resource Manager component of a AAA Server. We considered the problem of forwarding AAA messages among servers in possibly different administrative domains. We considered the need for end-to-end security of portions of the

payload of authorization messages that pass through intermediate AAA Servers. Finally we stressed the need for support of a streamlined authorization process that minimizes delay for latency-sensitive applications.

The intent is that this will provide support for discussing and understanding requirements of specific applications that need authorization services.

2.2.14 Security Considerations

Authorization is itself a security mechanism. As such, it is important that authorization protocols cannot easily be abused to circumvent the protection they are intended to ensure. It is the responsibility of protocol designers to design their protocols to be resilient against well-known types of attacks. The following are some considerations that may guide protocol designers in the development of authorization protocols.

Authorization protocols must not be susceptible to replay attacks. If authentication data is carried with the authorization data, for example, the authentication protocol used must either be impervious to replay or else the confidentiality of the authentication data must be protected.

If proxying is required, the authorization protocol must not be susceptible to man-in-the-middle attacks.

If the push model is used, the confidentiality of the authorization data must be ensured so that it may not be hijacked by third parties and used to obtain a service fraudulently.

If the agent model is used, the binding between the authorization and the service itself must be protected to prevent service authorized to one party from being fraudulently received by another.

In addition to guarding against circumvention, authorization protocols designed according to this framework will have some intrinsic security requirements. These are included among the requirements in [R2906] and summarized briefly below.

Among the intrinsic security needs is the fact that authorization protocols may carry sensitive information. It is necessary to protect such information from disclosure to unauthorized parties including (as discussed in section 2.2.11) even certain parties involved in the authorization decision.

We have discussed the use of multi-party trust chains involving relaying of authorization data through brokers or other parties. In such cases, the integrity of the chain must be maintained. It may be necessary to protect the data exchanged between parties using such mechanisms as encryption and digital signatures.

Finally, because an authorization server allow access to an Internet service, a denial of service attack targeted against an authorization server can be just as effective as a denial of service attack against the service equipment itself in preventing users to use a service.

2.3 The Generic AAA Architecture[2]

The Generic AAA Architecture [R2903] considers the architecture of a network of AAA servers acting as authority for the use of the resource(s) within its organisational domain.

We will first describe the envisioned Generic AAA Architecture goals before describing the architecture [I1471] itself. Sections 2.3.2 and 2.3.3 describe the architecture by considering the conceptual elements and its functional properties embodied in its elements and relationships. Section 2.3.4 will observe the key points of the model. Section 2.3.5 will consider suggestions for future work where elements spurred research of phase 2 (see section 1.3).

Sections 2.3.6 and 2.3.7 considers organizing the different Generic AAA functions. These sections have been included for completeness only. It discusses our thoughts at that time on how the protocol interactions and abstractions between the functional elements could be designed and constructed using a layered model. To serve this purpose the content of these sections have been reproduced in its unmodified form.

2.3.1 Generic AAA Architecture goals

As seen in the AAA Framework, authorization decisions spanning multiple organisations need to be taken by a network of AAA servers that interact with the User, AAA servers from other organisations (domains) and the Service Equipment residing within a domain. The envisioned long-term goal of the Generic AAA Architecture is to create a generic framework, which allows complex authorizations to be realized through a network of interconnected AAA servers, where the Generic AAA servers would communicate via a standard protocol. The protocol should be quite general and should support the needs of a wide variety of applications requiring AAA functionality. To realize this goal, the protocol will need to operate in a multi-domain environment with multiple service providers as well as entities taking on other AAA roles such as User Home Organizations and brokers. It should be possible to combine requests for multiple authorizations of different types in the same authorization transaction. The AAA infrastructure will be required to forward the components of such requests to the appropriate AAA servers for authorization and to collect the authorization decisions from the various AAA servers consulted. All of this activity is perfectly general in nature and can be realized in the common infrastructure.

---

2 This section is been based on:
RFC2903 "Generic AAA Architecture" C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, D. Spence, IETF August 2000.

Applications requiring AAA services will each have their own unique needs. After a service is authorized (by taking policy decisions), it must be configured and initialized. This initialisation process will require application specific knowledge and may require application specific protocols to communicate with application specific service components. To handle these application specific functions, the AAA Architecture Research Group proposed an application interface between a generic AAA server and a set of one or more Application Specific Modules (ASMs), which can carry out the unique functionality required by each application. In this section we will first consider the interactions of the generic AAA server with the Application Specific Modules and with each other to realize complex AAA functions.

The RFC also presents thought on how to organize the AAA functions into logical groups using a protocol layering model.

2.3.2 Generic AAA Server functional components

In this section we will describe what is needed to handle AAA requests in a generic, application independent way, where each stakeholder is able to have its own policy rules to take authorization decisions. The idea's regarding the functionalities described were envisaged at the start of our research. We will see these idea's develop during our research.

2.3.2.1 Evaluation of policy rules

The first step in the authorization process is for the user or an entity operating on the user's behalf to submit a well-formatted request to an AAA server. A generic AAA server has rules (logic and/or algebraic formulas) to inspect the request and come to an authorization decision. In order to be a generic rather than an application specific function, the first problem to consider is separating the Application Specific Information (ASI) from the underlying logic processing the authorization. Ideally, the AAA server would have a rule-based engine at this point, which would know the logic rules and understand some generic information in the request, but it would not know anything about the application specific information, except where this information can be evaluated to yield a Boolean (a yes or no) or a numerical value that can be compared. In this way it should be possible to create generic rules that refer to data elements that were not considered when the application, that combines services, was created. For example, a scientist could request to do an experiment by using a Network Service Provider that provides access to dedicated bandwidth needed to support his experiment. The request would only be successful if the Service Provider allows the User access to such facility (AAA1), the requested bandwidth is available from the intermediate networks (determined by Bandwidth Brokers governed by AAA2 and AAA3) and also if the User has the money to pay for using such facility (AAA4), after AAA2 and AAA3 have determined the cost.

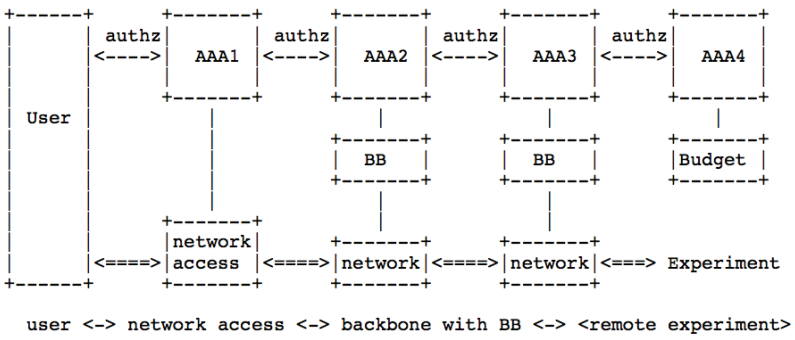


Fig. 2.3.1 -- Example of a Multi Domain Multi Type of Server Request

Possibly, the people who specified the Bandwidth Broker protocol did not think of combining Bandwidth requirements with a Network Service Access authorization and cost information in a single AAA request, but the intention of the generic model would be to allow it.

Note: Fig 2.3.1 shows a chain topology, but other topologies such as tree or star can also be examples. The OGF NSI [NSI] Working Group is nowadays working on various topologies

2.3.2.2 The Application Specific Module

Ultimately an AAA server needs to interact with an application specific module (ASM). In a service provider, the ASM would manage resources and configure the service equipment to provide the authorized service. It might also involve itself in the authorization decision because it has the application specific knowledge required. A user home organization (UHO) may require ASMs as well, to perform application specific user authorization functions. For example, a UHO ASM might be required to access certain application specific databases or interpret application specific service level specifications.

Whatever the role of an administration relative to an authorization decision, the capabilities of the generic AAA server and the interface between it and the ASMs remains the same. This interface may be an Application Program Interface (API) or could even be a protocol based interface. In this model, however, the application specific module is regarded as separate architectural component from the generic AAA server. As such, it must be addressable and must, therefore, be part of a global naming space.

2.3.2.3 Additional functional elements

Table 2.3.1 shows the functional elements were envisioned that could additionally be needed to support Generic AAA functions:

Functional Element	Description
Authorization Event Log	For auditing purposes, the generic server must have some form of database to store time-stamped events, which occur in the AAA server.
Policy Repository	A database containing the available services and resources about which authorization decisions can be made and the policy rules to make them is also needed. Here too, the naming space for the services and resources is important since they must be addressable from other AAA servers to be able to build complex authorization requests.
Request Forwarding	Due to the multiple administrative domain (multi-kingdom) nature of the AAA problem, a mechanism to forward messages between AAA servers is needed.

Table 2.3.1: Envisioned functional elements.

2.3.3 Generic AAA server model

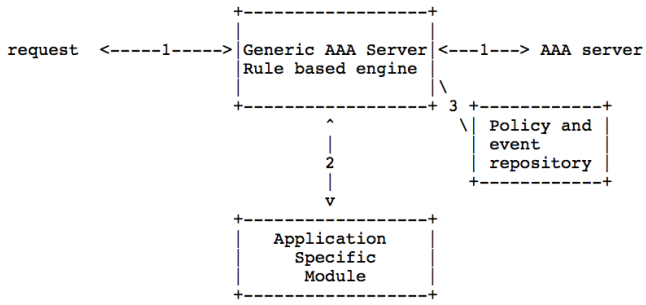
With the implementation of the above-mentioned components, the AAA server would be able to handle AAA requests. It would inspect the contents of the request, determine what authorization is requested, retrieve the policy rules from the repository, perform various local functions, and then choose one of the following options to further process each of the components of the request:

a) Let the component be evaluated by an attached ASM.
b) Query the authorization event log or the policy repository for the answer.
c) Forward the component(s) to another AAA server for evaluation.

In the following sections we present the generic model.

2.3.3.1 Generic AAA server interactions

Fig. 2.3.2 illustrates a generic AAA Server with connections to the various architectural components described above. In this model, the user or another AAA server contacts the AAA server to get authorization, and the AAA server interacts with the service. The request is sent to the AAA server using the future AAA protocol. The server interacts with the service via a second protocol which we have labelled as type "2" in the figure. We say no more of the type 2 protocol than that it must support some global naming space for the application specific items. The same holds for the type 3 communication used to access the repository.

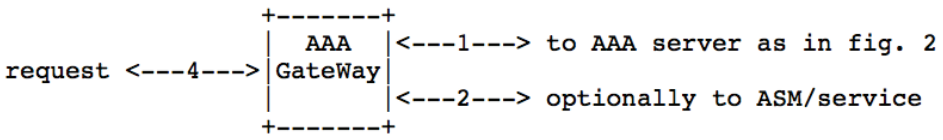


The numbers in the links denote types of communication.

Fig. 2.3.2-- Generic AAA Server Interactions

2.3.3.2 Compatibility with legacy protocols

Because of the widespread deployment of equipment that implements legacy AAA protocols and the desire to realize the functionality of the new AAA protocol while protecting the investment in existing infrastructure, it may be useful to implement a AAA gateway function that can encapsulate legacy protocol data units within the messages of the new protocol. Use of this technique, for example, would allow Radius attribute value pairs to be encapsulated in Application Specific Information (ASI) units of the new protocol in such a way that the ASI units can be digitally signed and encrypted for end-to-end protection between a service provider's AAA server and a home AAA server communicating via a marginally trusted proxy AAA server. The service provider's NAS would communicate via Radius to the service provider's AAA server, but the AAA servers would communicate among themselves via the new AAA protocol. In this case, the AAA gateway would be a software module residing in the service provider's AAA server. Alternatively the AAA gateway could be implemented as a standalone process. Figure 2.3.3 illustrates an AAA gateway. Communication type 4 is the legacy protocol. Communication type 1 is the future standard AAA protocol. And communication type 2 is for application specific communication to Application Specific Modules (ASMs) or Service Equipment.



The numbers in the links denote types of communication.

Fig. 2.3.3 -- AAA Gateway for Legacy AAA Protocols

2.3.3.3 Interactions between the ASM and the Service

In a service provider, the Application Specific Module (ASM) and the software providing the service itself may be tightly bound into a single "Service Application". In this case, the interface between them is just a software interface. But the service itself may be provided by equipment external to the ASM, for example, a router in the bandwidth broker application. In this case, the ASM communicates with the service via some protocol. These two possibilities are illustrated in figure 2.3.4. In both cases, we have labelled the communication between the ASM and the service as communication type 5, which of course, is service specific.

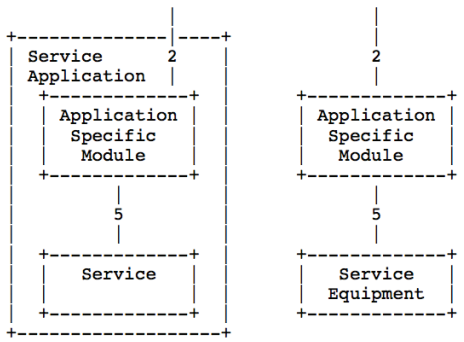
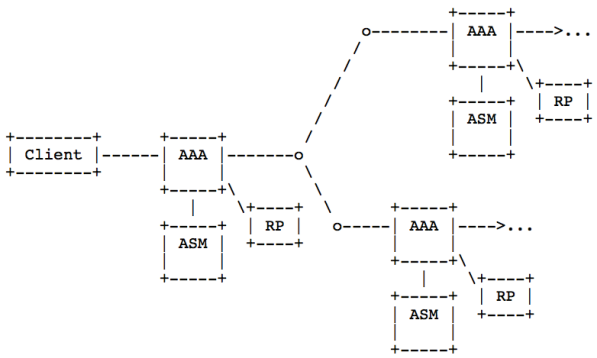


Fig. 2.3.4 -- ASM to Service Interaction (two views)

2.3.3.4 Multi-domain Architecture

The generic AAA server modules can use communication type 1 to contact each other to evaluate parts of requests. Figure 2.3.5 illustrates a network of generic AAA servers in different administrative domains communicating via communication type 1.



The AAA servers use only communication type 1 to communicate.
ASM = Application Specific Module
RP = Repository

Fig. 2.3.5 -- Multi-domain Multi-type of Service Architecture

2.3.4 Model Observations

Some key points of the generic architecture are:

1) The same generic AAA server can function in all three authorization framework models recognized in RFC2904: agent, pull, and push.
2) The rule based engine knows how to evaluate logical formulas and how to parse AAA requests.
3) The Generic AAA server has no knowledge whatsoever about the application specific services, so it is opaque to the application specific information it forwards.
4) Communication types 1, 2, and 3 each present their own naming space problems. Solving these problems is fundamental to forwarding AAA messages, locating application specific entities, and locating applicable rules in the rule repositories.
5) A standard AAA protocol for use in communication type 1 should be a peer-to-peer protocol without imposing client and server roles on the communicating entities.
6) A standard AAA protocol should allow information units for multiple different services belonging to multiple different applications in multiple different administrative domains to be combined in a single AAA protocol message.

2.3.5 Suggestions for future work on Generic AAA.

The following suggestions for future work were envisaged after the initial work on Generic AAA was completed.

It is hoped that by using this generic model it will be feasible to design an AAA protocol that is "future proof", in a sense, because much of what we do not think about now can be encoded as application specific information and referenced by policy rules stored in a policy repository. From this model, some generic requirements arise that will require some further study. For example, suppose a new user is told that somewhere on a specific AAA server a certain authorization can be obtained. The user will need an AAA protocol that can:

1) send a query to find out which authorizations can be obtained from a specific server,
2) provide a mechanism for determining what components must be put in an AAA request for a specific authorization, and
3) formulate and transmit the authorization request.

Some areas where further work is particularly needed are in identifying and designing the generic components of a AAA protocol and in determining the basis upon which component forwarding and policy retrieval decisions are made.

In addition to these areas, there is a need to explore the management of rules in a multi-domain AAA environment because the development and future deployment of a generic multi-domain

AAA infrastructure is largely dependent on its manageability. Multi-domain AAA environments housing many rules distributed over several AAA servers quickly become unmanageable if there is not some form of automated rule creation and housekeeping. Organizations that allow their services to be governed by rules, based on some form of commercial contract, require the contract to be implemented with the least possible effort. This can, for example, be achieved in a scalable fashion if the individual user or user organization requesting a service is able to establish the service itself. This kind of interaction requires policy rule establishment between AAA servers belonging to multiple autonomous administrative domains.

2.3.6 Layered AAA Protocol Model

Note: this section has been included for completeness only. The idea's presented may not have been implemented or implemented differently in our subsequent research.

In the previous section, we proposed the idea of a generic AAA server with an interface to one or more Application Specific Modules (ASMs). The generic server would handle many common functions including the forwarding of AAA messages between servers in different administrative domains. We envision message transport, hop-by-hop security, and message forwarding as clearly being functions of the generic server. The application specific modules would handle all application specific tasks such as communication with service equipment and access to special purpose databases. Between these two sets of functions is another set of functions that presumably could take place in either the generic server or an ASM or possibly by a collaboration of both. These functions include the evaluation of authorization rules against data that may reside in various places including attributes from the authorization request itself. The more we can push these functions down into the generic server, the more powerful the generic server can be and the simpler the ASMs can be.

One way of organizing the different functions mentioned above would be to assign them to a layered hierarchy. In fact, we have found the layer paradigm to be a useful one in understanding AAA functionality. This section explores the use of a layered hierarchy consisting of the following AAA layers as a way of organizing the AAA functions:

- Application Specific Service Layer
- Presentation Service Layer
- Transaction/Session Management Service Layer
- Reliable/Secure Transport Service Layer

Nevertheless, the interface between the generic AAA server and the ASMs proposed in the previous section may be more complex than a simple layered model would allow. Even the division of functionality proposed in this section goes beyond a strict understanding of layering. Therefore, this RFC can probably best be understood as the beginnings of a work to understand and organize the common functionality required for a general purpose AAA infrastructure rather than as a mature reference model for the creation of AAA protocols.

In our view of AAA services modelled as a hierarchy of service layers, there is a set of distributed processes at each service layer that cooperate and are responsible for implementing that service layer's functions. These processes communicate with each other using a protocol specialized to carry out the functions and responsibilities assigned to their service layer. The protocol at service layer n communicates to its peers by depending on the services available to it from service layer n-1. The service layer n also has a protocol end point address space, through which the peer processes at service layer n can send messages to each other. Together, these AAA service layers can be assembled into an AAA protocol stack.

The advantage of this approach is that there is not just one monolithic "AAA protocol". Instead there is a suite of protocols, and each one is optimized to solve the problems found at its layer of the AAA protocol stack hierarchy. This approach realizes several key benefits:

- The protocol used at any particular layer in the protocol stack can be substituted for another functionally equivalent protocol without disrupting the services in adjacent layers.
- Requirements in one layer may be met without impact on protocols operating in other layers. For example, local security requirements may dictate the substitution of stronger or weaker "reliable secure transport" layer security algorithms or protocols. These can be introduced with no change or awareness of the substitution by the layers above the Reliable/Secure Transport layer.
- The protocol used for a given layer is simpler because it is focused on a specific narrow problem that is assigned to its service layer. In particular, it should be feasible to leverage existing protocol designs for some aspects of this protocol stack (e.g. CORBA GIOP/CDR for the presentation layer).
- A legacy AAA protocol message (e.g. a RADIUS message) can be encapsulated within the protocol message(s) of a lower layer protocol, preserving the investment of a Service Provider or User Home Organization in their existing AAA infrastructure.
- At each service layer, a suite of alternatives can be designed, and the service layer above it can choose which alternative makes sense for a given application. However, to ensure some minimal functionality, it should be a primary goal of the AAA protocol standardization effort to specify one mandatory to implement protocol at the AAA Transaction/Session Management (AAA-TSM) service layer.

2.3.6.1 Elements of a layered architecture.

At each layer of a layered architecture, a number of elements need to be defined. These elements are shown in table 2.3.2.

Element	Description
Service Layer Abstract Interface Primitives	The service layer n is assumed to present a program interface through which its adjacent service layer n+1 can access its services. The types of abstract program service primitives and associated parameters exchanged across the boundary between these service layers must be specified.
Service Layer Peer End Point Name Space	Each service layer is treated as a set of cooperating processes distributed across multiple computing systems. The service layer must manage an end point name space that identifies these peer processes. The conventions by which a service layer assigns a unique end point name to each such peer process must be specified.
Peer Registration, Discovery, and Location Resolution	Along with defining an end point name space, a service layer must also specif1y how its peers: • announce their presence and availability, • discover one another when they first begin operation, and • detect loss of connectivity or service withdrawal. It is also necessary to specify what mechanisms, if any, exist to resolve a set of service layer specific search attributes into one or more peer end point names that match the search criteria.
Trust Relationships Between Peer End Points	Once an end point has established its initial contact with another peer, it must decide what authentication policy to adapt. It can trust whatever authentication was done on its behalf by a lower service layer or, through a pre-provisioning process, implicitly trust the peer, or else go through an authentication process with its peer. The supported and available mechanisms for establishing a service layer's end point trust relationships must be indicated at initial contact.
Service Layer Finite State Machine	To the extent that a service layer's internal states are externally visible, the layer's behaviour in terms of a Finite State Machine (FSM) should be specified. Events that can drive the FSM state transitions may include: • service layer n+1 interface primitive requests • protocol data unit arrivals from peer service layer n end points received through the layer n-1 access point • service layer n-1 interface primitives (e.g. call backs or interrupts) timer expirations
Protocol Data Unit Types	Each service layer defines a lexicon of protocol data units (PDUs) that communicate between the layer's peer processes the information that controls and/or monitors that service layer's distributed state and allows the service processes of that layer to perform their functions. Embedded in the PDUs of each layer are the PDUs of the higher layers which depend on its services. The PDUs of each service layer must be specified.

Table 2.3.2: Definitions of elements of a layered AAA architecture

2.3.6.2 AAA Application Specific Service Layer

AAA applications have almost unlimited diversity, but imposing some constraints and commonality is required for them to participate in this generic AAA architectural framework. To satisfy these constraints, participating AAA applications would derive their application specific program logic from a standardized "Authorization Server" abstract base object class. They would also support an "Authorized Session" object class. An Authorization Session object instance represents an approved authorization request that has a long-lived allocation of services or resources. The generic AAA architecture could be extended to include other abstract base object classes in the future (e.g. Authorization Reservation, Authentication Server, etc.). How to implement the derived Authorization Server class's public methods for a given problem domain is entirely up to the application. One technique might be to place a software "wrapper" around an existing embedded application specific service to adapt it to the standardized Authorization Server object paradigm. The major Authorization Server class methods are:

- Publish an advertisement that describes the Authorization Server's service attributes and its application specific service layer end point address. Once the Authorization Server has registered, peer processes can discover its presence or send messages addressed to it.
- Application Specific Authorization Decision Function (AS-ADF) method takes a User's application specific authorization request and returns a decision of approve, deny, or conditionally approve with referral to another stakeholder. In the latter case, the application may create a reservation for the requested services or resources. This method represents the "condition" side of a policy rule's condition/action pair.
- Commit a service or set of resources to a previously conditionally approved authorization decision. For those authorization requests that have a long-term lifecycle (as opposed to being transactions), this method mobilizes a reservation into an Authorized Session object instance. This method represents the "action" side of a policy rule's condition/action pair.
- Cancel a previously conditionally approved Authorization request. This method releases any associated reservations for services or resources.
- Withdraw the Authorization Server's service advertisement.

A key motivation for structuring an AAA application as an Authorization Server object instance is to separate the generic authorization decision logic from the application-specific authorization decision logic. In many cases, the application can be divorced from the AAA problem altogether, and its AAA responsibility can be assigned to an external rules based generic AAA Server. (The idea is similar to that of a trust management policy server as defined in [R2704].) This would facilitate a security administrator deploying AAA policy in a central repository. The AAA policy is applied consistently across all users of the applications, resources, and services controlled by the AAA server. However, it is recognized that for many problem domains, there are unique rules intrinsic to the application. In these cases, the generic AAA Server must refer the User's authorization request to the relevant Application Specific Module.

2.3.6.3 Presentation Service Layer

The presentation service layer solves the data representation problems that are encountered when communicating peers exchange complex data structures or objects between their heterogeneous computing systems. The goal is to transfer semantically equivalent application layer data structures regardless of the local machine architecture, operating system, compiler, or other potential inter- system differences.

One way to better understand the role of the presentation layer is to evaluate an existing example. The Generic Inter-ORB Protocol (GIOP) and its Common Data Representation (CDR) is a presentation service layer protocol developed by the Object Management Group (OMG) industry consortium. GIOP is one component within the Common Object Request Broker Architecture (CORBA). Peer Object Request Brokers (ORB) executing on heterogeneous systems use GIOP to invoke remote CORBA object interface methods. GIOP encodes an object method's input and output parameters in the Common Data Representation (CDR). While there are other presentation service layer protocols in the industry, GIOP in combination with CDR represents a mature, comprehensive solution that exhibits many of the presentation service layer requirements that are applicable within the AAA protocol model.

In the context of Internet access AAA protocols, RADIUS and its successors use the Attribute Value Pair (AVP) paradigm as the presentation service layer encoding scheme. While such an approach is versatile, it is also prone to becoming splintered into many ad hoc and vendor specific dialects. There is no structure imposed or method to negotiate the constraints on which AVPs are combined and interpreted for a given conversation in a consistent way across AAA protocol implementations or problem domains. At run-time, it can be hard for the communicating peers to negotiate to a common inter-operable set of AVPs.

To avoid this pitfall, a primary presentation service layer responsibility is the ability to let peers negotiate from a base Authorization Server object class towards a commonly understood derived Authorization Server object class that both presentation service layer peers have implemented for their application specific problem domain. This negotiation implies a requirement for a globally registered and maintained presentation service layer hierarchy of Authorization Server object class names.

2.3.6.4 AAA Transaction/Session Management Service Layer

The AAA Transaction/Session Management (AAA-TSM) service layer is a distributed set of AAA Servers, which typically reside in different administrative domains. Collectively they are responsible for the following three services:

- **Authentication** Execute the procedure(s) needed to confirm the identity of the other parties with which the AAA TSM entity has a trust relationship.

- **Authorization** Make an authorization decision to grant or deny a User's request for services or resources. The generic rules based policy engine described earlier in this document executes the authorization decision function. When the User's request is instantaneous and transient, then its authorization approval is treated as an ephemeral transaction. If the authorization approval implies a sustained consumption of a service or resources, then the request is transformed into an Authorized Session. For the duration of the Authorized Session's lifetime:
    - its state may be queried and reported, or
    - it may be cancelled before service is completed, or
    - the service being delivered may be modified to operate under new parameters and conditions, or
    - the service may complete on its own accord.

  In each of these cases, the AAA-TSM service layer must synchronize the Authorized Session's distributed state across all of those AAA Servers which are implementing that specific Authorized Session.
- **Accounting** Generate any relevant accounting information regarding the authorization decision and the associated Authorized Session (if any) that represents the ongoing consumption of those services or resources.

The peer AAA servers and their AAA-TSM end points exchange AAA-TSM messages to realize these AAA functions. A central AAA-TSM concept is that there is a set of one or more AAA Server stakeholders who are solicited to approve/disapprove a User request for application layer services. The AAA-TSM service layer routes the User's request from one stakeholder to the next, accumulating the requisite approvals until they have all been asked to make an authorization decision.

The AAA Servers may also do User authentication (or re-authentication) as part of this approval process. The overall flow of the routing from one stakeholder to another may take the form of the "push", "pull", or "agent" authorization models developed in [R2904]. However, in principle, it is feasible to have an arbitrary routing path of an AAA-TSM authorization request among stakeholders. Once the final approval is received, the AAA-TSM service layer commits the requested service by notifying all of those stakeholders that require a confirmation (i.e. turn on a pending reservation and do a transaction commit). Alternatively, any stakeholder among those on the consent list can veto the authorization request. In that case, all stakeholders who previously approved the request and had asked for a confirmation are told that the request has been denied (i.e., cancel reservation and do a transaction rollback).

The AAA-TSM authorization request payload must carry its own "Context State", such that when an AAA server receives it, there is sufficient information that it is essentially self-contained. Embedding the Context State within the AAA-TSM message provides two benefits. First, the message can be immediately processed with respect to the AAA Server's local policy, and this minimizes or altogether avoids the need for the AAA Server to exchange additional AAA-TSM messages with its peers to complete its piece of the overall authorization decision. The other

benefit is that the AAA Server minimizes the amount of state information resources that it commits to a user's pending request until it is fully approved. This helps protect against denial of service attacks.

One can envision many possible message elements that could be part of the Context State carried within an AAA-TSM request message:

- AAA-TSM session identifier, a unique handle representing this authorization request. All AAA servers who participate in a request's approval process and its subsequent monitoring throughout its Session lifetime refer to this handle.
- permission lists stating which AAA Servers are allowed to modify which parts of the message.
- User's authorization request, encoded as a presentation layer PDU.
- User authentication information, (e.g. an X.509 public key certificate).
- User credentials information, or else a pointer to where that information can be found by an AAA server. An example of such credentials would be an X.509 attributes certificate.
- the list of AAA Server stakeholders who have yet to be visited to gain full approval of the User's authorization request. Each element in that list contains a presentation layer message encoding how the user authorization request should be evaluated by its application specific Authorization Decision Function (ADF).
- the current position in the list of AAA Server stakeholders to be visited.
- a list of those AAA servers which have already conditionally approved the User's authorization request, but which have predicated their approval on the request also completing its approval from those stakeholders who have not yet seen the request. Each element in the list has a digital signature or comparable mechanism by which their approval can be subsequently verified.
- an expiration time stamp, expressed in a universally understood time reference, which sets a lifetime limit on the AAA-TSM message's validity. This offers some replay attack protection, and inhibits messages from circulating indefinitely seeking the completion of a request's approval.
- a message payload modification audit trail, tracing which parties introduced changes into the User's authorization request terms and conditions.
- an AAA-TSM message integrity check, computed across the whole message rather than its individual elements, and signed by the most recent AAA-TSM layer end point process to modify the AAA-TSM message before its transmission to its AAA-TSM peer. This function may be delegated to the underlying Reliable Secure Transport layer connection to that destination peer.

2.3.6.5 Service Layer Program Interface Primitives

The AAA-TSM service layer and its adjacent presentation service layer communicate across their boundary through a set of program interface primitives. A key design goal is to keep

these primitives the same regardless of the higher level AAA application, analogous to a callable "plug-in". The two service layers are responsible for coordinating their state information. This responsibility includes all of the pending Authorization requests and the Authorization Sessions that they are both controlling and monitoring. The initial contact between these two layers is through an abstract object that is called an AAA-TSM Service Access Point (SAP). A particular service instance between these two layers is realized in an abstract object that is called an Authorized Session. The presentation service layer invokes AAA-TSM interface primitives against an AAA-TSM SAP.

The AAA-TSM service layer interface primitives can be broadly characterized as follows:

- Send a presentation layer message to a specified destination presentation layer peer end point address.
- Receive a presentation layer message from another presentation layer end point address. A receive operation may select a specific originating presentation layer end point address from which the message is expected, or receive a message from any presentation layer peer.
- The AAA-TSM service layer calls an application specific authorization decision function, which returns a condition code expressing an approval, denial, or partially approves with a referral to another AAA Server.
- AAA-TSM service layer tells the presentation layer to commit an earlier partially approved authorization request.
- Cancel an earlier partially approved authorization request (i.e. rollback).
- The presentation service layer notifies the AAA-TSM service layer that it has terminated an in-progress Authorized Session.
- AAA-TSM service layer notifies the presentation service layer that another presentation service layer peer has terminated an Authorized Session.
- Un-register a presentation service layer end point address.

2.3.6.6 Service Layer End Point Name Space

The AAA-TSM service layer end point name space is the N-tuple formed by concatenating the following components:

- AAA Server's Reliable/Secure Transport layer end point address.
- AAA-TSM authorization request serial number, a unique durable unsigned integer generated by the AAA Server who first receives the User's authorization request.

Some AAA applications may require that each assigned AAA-TSM transaction serial number be stored in persistent storage, and require that it be recoverable across AAA Server system re-boots. The serial number generation algorithm must be guaranteed unique even if the AAA Server does a re-boot.

2.3.6.7 Protocol Stack Examples

The layering paradigm makes it possible to use the most appropriate syntax for each application for encoding the Application Specific Information units of that application. This encoding would take place at the presentation layer. Similarly the application layer can recognize the semantics specific to each application. Figure 2.3.6 illustrates some possible AAA protocol stacks.

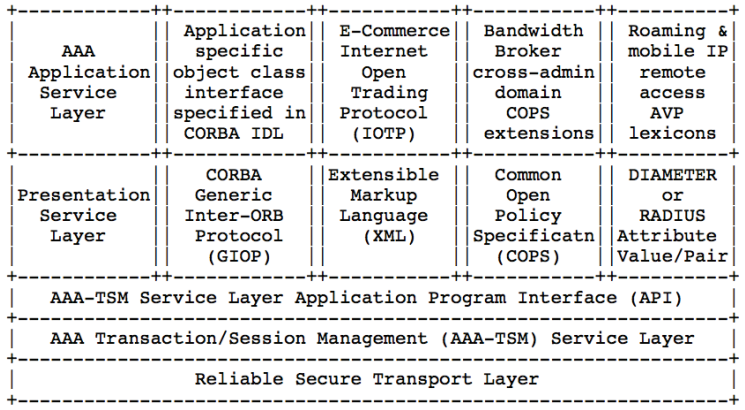


Fig. 2.3.6 -- Possible AAA Protocol Stacks

2.3.7 Security Considerations

Security considerations for the framework on which the work described in this memo is based are discussed in [R2904]. Security requirements for authorization are listed in section 2.2 of [R2905].

This memo identifies a basic set of AAA functions that are general in nature and common to many different AAA applications. We propose that a standard set of security mechanisms should be defined as part of a base AAA protocol which would include such things as public key encryption and digital signatures that could be applied to individual information units within an AAA message. Security with this granularity is needed to meet the end-to-end security requirement specified in section 2.2.7 of [R2905] because a single AAA message may contain multiple information units each generated by AAA servers from different administrative domains and destined to AAA servers in different domains.

In addition, it may be necessary to encrypt or sign an entire AAA message on a hop-by-hop basis. This could be handled by a standard, lower layer protocol such as IPSEC. If so, then certain auditing requirements will have to be met so that it can be established later that the messages relative to some specific session ID were, in fact, protected in a particular way. Alternatively, hop-by-hop security mechanisms may be built into the base AAA protocol itself.

2.4 Summary

In section 2.1 we have seen different technologies in a number of contexts, which can be used to construct authorization solutions. Pre-historic technologies such as sealing and stamping a clay envelope containing tokens as message has for example its Internet equivalent called HTTPS, which secures messages between browsers and servers handling of web pages and to ensure that a user has a connection with the intended website. Rules determine what the content of a message means at business side. A clay sphere may mean one bowl of cereal has been contributed as tax when contained in an envelope stamped by a recognized authority. Pushing a button on a website may mean that a customer authorizes a website to initiate a payment transaction as part of buying goods. In both cases, a common understanding of the meaning of the elements of an authorization message is as important as understanding the policy driven process taking authorization decisions as well as the mechanisms that are used to secure a message. The overall understanding between all involved parties to handle a transaction securely and correctly is an essential need to create trust in its operation. We have seen many examples of mechanisms that address the security of handling transaction messages, but what does it mean to create trust by handling transactions correctly? This question lead to sub question 4:
What is needed to arrange trust when authorizing e-infrastructure resources?

In section 2.2 we have seen a way to articulate scenarios amongst a number of entities that take part in an authorization transaction. Here we recognized a number conceptual interaction patterns: The push-, pull- and agent sequence model and its combinations are expected to be useful as a generic way to describe authorization scenarios in different cases. The section also identifies the need to make trust an explicit part of handling authorization transactions and contributed to asking sub question 4.

In section 2.3 we have seen an architecture comprising of a number of functional elements that can communicate authorization transactions in a network of authorization servers. The Rule Based Engine in an AAA server can take policy decisions that are retrieved from a policy repository depending on the message received. When invoked, the policy can call application specific functions that are performed by Application Specific Modules (ASM). These modules allow AAA services to interact with the outside world, e.g. configure devices or communicate with other AAA servers. This architecture shows how a network of AAA servers is expected to work in multi-domain scenarios. This architecture, however, needs to be verified to prove its applicability. This lead to sub question 2:
What generic authorization concepts are expected to work best for classes of applications that use multi-domain network resources?
And sub question 3:
How can we apply the generic multi-domain authorization concepts in Network QoS / Lightpath provisioning class of applications?

2.5 Evolution and contributions

The presented research, which evolved over a period of fifteen years, resulted in a number of publications used as basis for this thesis. As mentioned in section 1.3, this period can be divided into three phases as shown in fig 2.5.1. In each of the three phases, the author studied multi-domain authorization cases in a specific problem context. Next sections will explain the approach of each phase and the author's contributions. In sections 2.2 until 2.4 we saw the results of phase 1. As an introduction to chapters 3 and 4 (phase 2) and chapter 5 (phase 3) we will now explain the context and approach in more detail.

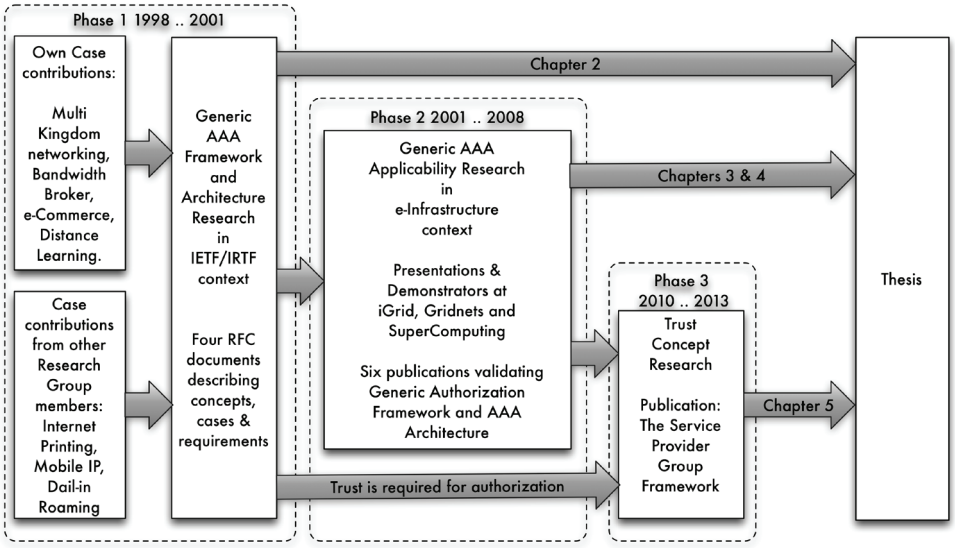


Fig 2.5.1: This thesis comprises research over a 15-year period, which can be subdivided into three phases. Each phase resulted in publications that contribute to this thesis.

2.5.1 Phase 1 Generic AAA concepts

As we saw, the multi-kingdom question was first taken to the Internet Engineering Task Force [IETF]. The IETF is a community working on standards and technologies to support the development of the Internet infrastructure. Here suitable work was explored, in particular in the area of Authentication, Authorization and Accounting (AAA). Authorization related technologies and additional cases were found that resembled the multi-kingdom case on hand. However, no approach was found that would be generically applicable. The IETF therefore allowed the question to be placed in the context of an Internet Research Task Force [IRTF] Research Group [AAAARG]. Based on a number of cases and identified technology gaps, an AAA

Authorization Framework and Generic AAA Architecture were developed. Both the Framework and Architecture describe concepts that, based on validation when applied to a number of cases [R2905], were expected to handle authorization transactions generically. In collaboration with Internet pioneer Merit [MERIT], Interpay [ITP] and University of Utrecht, the author focussed on identifying and modelling authorization sequences and an architecture that could be applied to cases such as bandwidth brokerage [IETF45, QBON] (fig 2.5.2), e-commerce and computer based distance learning. These scenarios included authorization transactions arranging access to resources contributed by multiple domains. Other research group members worked on cases such as Internet Printing, Mobile-IP and Dial-In roaming. These members verified the generic model, concepts and terminology by using them in describing their cases. However, to disentangle the question from specific technical cases, we decided to always keep a more generic question in the back of our mind: *"What would be needed to handle a request to deliver an online movie, the bandwidth to transport the movie over the Internet and a pizza to go along with watching the movie?"*. As will be explained, existing authorization methods at that time (such as Role Based Access Control [FERR]) were less suitable to handle such authorization sequences.

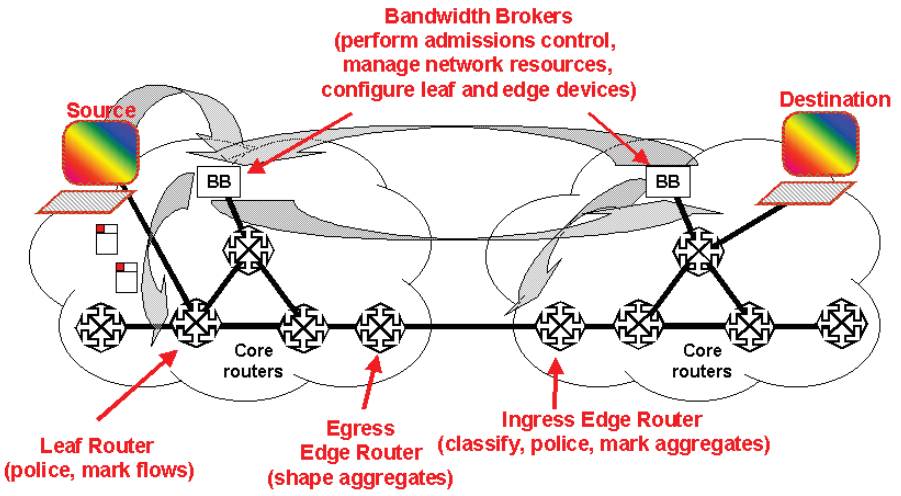


Fig 2.5.2: AAA scenarios for bandwidth brokerage providing Quality of Service networking formed an important initial research case that resembled the multi-kingdom case. A bandwidth broker configures routers to provide a different forwarding behaviour to IP packet flows that contain special markers in their packet headers.

The research work was made explicit in Request for Comment (RFC) documents as part of the IETF community output. By considering a number of different applications, the documents suggest that the generic approach is expected to be suitable to handle several types of authorization transaction scenarios.

RFC	Description
RFC2903	The Generic AAA Architecture [R2903], describing a set of elementary functional elements that are capable of handling authorizations in a distributed way. It separates the logic of handling requests from the semantics of a request. It also describes a number of protocol types that can be recognized between these functions.
RFC2904	The Authorization Framework [R2904], describing a way to think about modelling authorization sequences between elementary functional elements. It describes three fundamental sequences between these functions and its use in several scenarios.
RFC2905	The AAA Authorization Application Examples [R2905], explaining a number Internet applications requiring a generic authorization mechanism. These examples were used in order to understand their requirements. It uses the Authorization Framework concepts to model these applications.
RFC2906	The AAA Authorization Requirements [R2906], listing the requirements found in the study performed on the Application Examples.

Table 2.5.1: Output of phase 1 IRTF research work.

2.5.2 Phase 2: Generic AAA applicability

The applicability of the Generic AAA concepts needed to be validated within a community providing suitable scenarios. The Internet Society recognizes [ISOC] that *"the Internet is as much a collection of communities as a collection of technologies, and its success is largely attributable to both satisfying basic community needs as well as utilizing the community in an effective way to push the infrastructure forward"*. The global e-Science community, using applications that process and move data at peta-byte scale, can be seen as an example community pushing the Internet infrastructure forward in order to satisfy its basic community needs. As explained in the introduction, we saw e-Infrastructures [EIRG] increasingly enabling research carried out through distributed regional, national and global collaborations. Within such collaborations, data is growing at exceptional rates [BERI] and requires rethinking of the ways it is handled [HEY] and transported [TIER, BROW]. In 2001, the use of optical network technology, to support e-Science class of applications, was seen as an opportunity by de Laat et. al. [DLA3]. He explains that optical networking can avoid expensive routing, in particular when such applications only need a few locations to be inter-connected. Moreover, transportation of such large data volumes would be a disturbing event for the regular Internet. Although already recognized at that time, it is still recognized now [ARNA] that such application driven transport poses both technical and business challenges, in particular when such connections are to be arranged across multiple network providers via exchanges. An important recognition [DLA3] back then was *that dynamic setup and teardown of optical circuits will only be used if adequate policy systems were developed and installed to control these new resources.* This important observation motivated us to study the applicability of the Generic AAA concepts as a contribution towards

the needs of e-Infrastructures. Providing researchers with access to dedicated network resources is a responsibility of NRENs. Therefore, the NREN community represented a suitable context to experiment with the applicability of the Generic AAA Framework and Architecture.

Fig 2.5.3: Tiled displays are used to create high resolution visualisation facilities. Here such a setup was demonstrated by SARA at SuperComputing 2005 in Seattle showing hi-resolution images from servers located in Amsterdam.

NRENs support network research work performed in standards bodies and international collaborative projects. The Open Grid Forum [OGF] and Global Lambda Integrated Facility [GLIF] are communities that work on ways to handle data intensive e-Science applications. The OGF works on the Grid computing, a concept that processes data by scheduled batches of parallel processes. Such applications may need scheduled or on-demand use of dedicated multi-gigabit network circuits to transfer data between collaborating Grid computing data centres and/or large experiments, visualisation facilities (fig 2.5.3), etc. The GLIF consortium promotes the dynamic transport of large data volumes at worldwide scale. It uses optical high bandwidth circuits and network exchanges [NLIG, STARL] (see fig. 2.5.4) provided by Internet pioneering NREN organisations such as Internet2, SURFnet, ESNet, Gloriad, Canarie, Nordunet, and more [GLIFA]. By means of a simple agreement, the GLIF arranges the redistribution of the surplus Internet transport capacity of participating NRENs. In this way the GLIF is able to support e-Science communities such as LHCOPN [CERN], OptIPuter [SMARR, PIEP] and more [GLIFA]. Next to GLIF, other communities such as the Global Environment for Network Innovations [GENI], Internet2's Dynamic Circuit Network / Advanced Layer 2 Service [DCN] and the OGF's Network Service Interface Working Group [NSI] are working on ways to provide what is now called end-to-end Lightpath support. The OGF and GLIF communities, therefore, represent an important source of authorization scenarios for the use of Lightpaths.

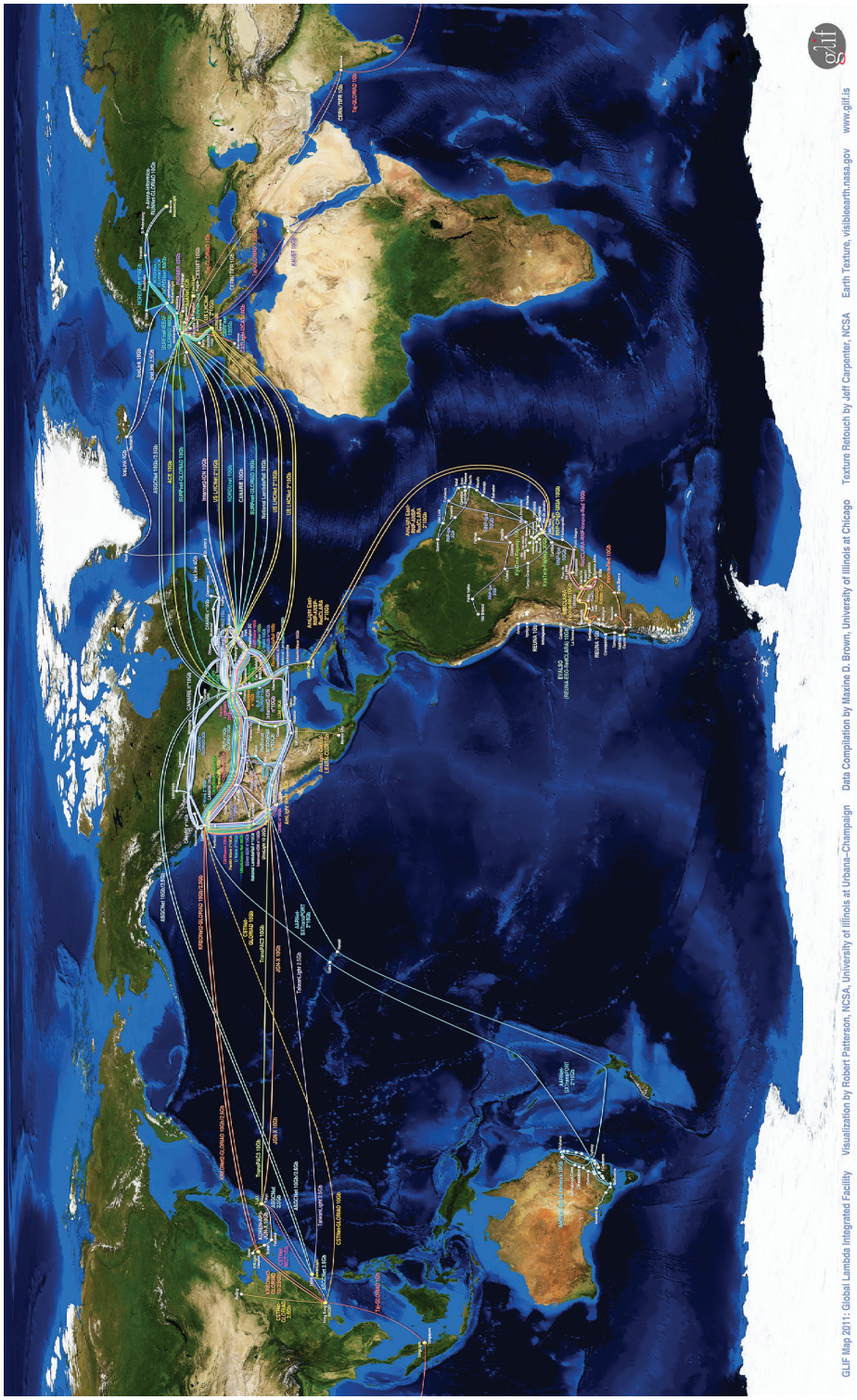


Fig. 2.5.4: The Global Lambda Integrated Facility, supporting e-Science communities by providing optical network resource that can be chained into end-to-end connections.

The author's contributions in researching the applicability of the Generic AAA concepts have been described in a number of publications shown in table 2.5.2. These contributions validate the applicability in the context of the automatic creation dedicated network connections as part of e-Infrastructures. A "Generic AAA Toolkit" [AAATK] was developed to implement the concepts. The first two publications validate the applicability of the so called "agent model", where agents are placed in a network making distributed authorization decisions (see example in fig 2.5.5). The subsequent publications consider the "token" model. In this model an agent hands a token back to the application for subsequent use in the infrastructure. This model was validated by implementations inserting tokens at different network technology layers. The concepts were demonstrated at different events. Fig 2.5.6 shows for example the demonstration at SuperComputing 2007.

	Contributing publication	**Demonstrated (D) and/or Presented (P) at**
1	Authorization of QoS path based on Generic AAA [GOM3]	iGrid 2002 (D)
2	Applications Drive Secure Lightpath Creation across Heterogeneous Domains [GOM61]	SuperComputing 2004 (D)
3	Token based authorization of Connection Oriented Network resources [GOM4]	Gridnets 2004 (P)
4	Token Based path authorization at Interconnection Points between Hybrid Networks and a Lamda Grid [GOM5]	Gridnets 2005 (D)
5	Token Based Networking: Experiment NL101 [GOM62]	iGrid 2005 (D)
6	Multi-domain lightpath authorization, using tokens [GOM8]	SuperComputing 2006 (D) SuperComputing 2007 (D)

Table 2.5.2 The author's publications comprising the outcome of phase 2, which were published after a demonstration or important conference presentation [IGRI, SCOR].

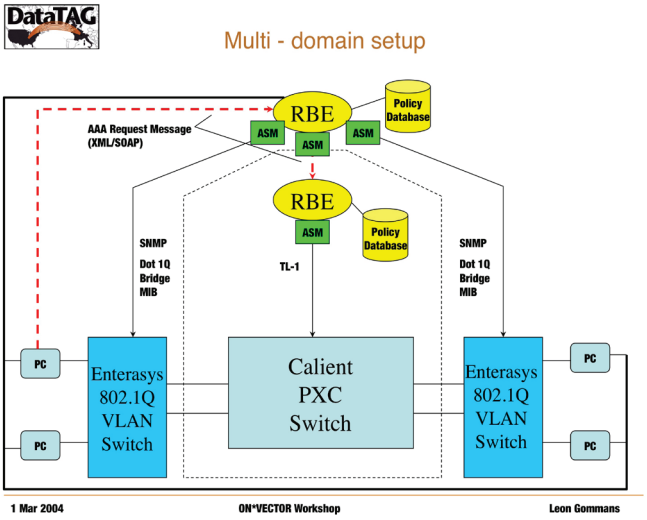


Fig 2.5.5 One of the early experiments with the Generic AAA toolkit components (RBE=Rule Based Engine, ASM=Application Specific Module) acting as controlling agents for VLAN switches and a photonic cross connect switch (PXC), switching optical connections using micro-electomechanical mirror switch technology. The effort was a contribution to the EU DataTAG project.

This demonstration, performed together with Internet2, showed that tokens could be used to authorize the viewing of an uncompressed HD movie and the necessary bandwidth to Reno as it was streamed from Amsterdam.

Fig 2.5.6: Our demonstration booth at SuperComputing 2007 in Reno, Nevada. This demonstration was performed in collaboration with Internet2's Dynamic Circuit Network demonstration.

In phase 2, the author has contributed in the area of AAA research to a number of Network Research projects [LG], in particular to the national SURFnet GigaPort [GIGA] projects, the EU DataTAG [DATA], DataGRID [DAGR], NextGRID [NEXT] and Phosphorous [PHOS] projects and the International OptiPuter project [OPTI], the GLIF/LambdaGrid [LAMB] , the Internet2 DCN/ION [DCN] project and ESNet Dragon [DRAG] project.

Within the OGF, the author contributed to the work of the Authorization Frameworks and Mechanisms Working Group (AuthZ-WG) [OGSA, GFD38], the Grid High Performance Networking Research Group (GHPN-RG) [GHPN, GOM62] and by co-chairing the Firewall Issues Research Group (FI-RG) [FIRG, GFD83, GFD142].

2.5.3 Phase 3: Trust concept research

Trust plays an important role in allowing authorizations to happen. This important observation was noted in RFC2904. Phase 3 contributes towards a better understanding of the concept of trust in relation to authorization. As authorization is based on the execution of policies, the research in phase 1 asserted that trust is necessary to allow each entity to know that the policy it is authorizing is correct. Understanding correctness is both a business issue as well as a protocol issue. Within small groups that own few resources, authorization and trust can be based on informal personal understanding, which is used to configure access by hand on devices providing the service. Science communities traditionally created such small-scale solutions that work well. In modern society, however, many global authorisation transactions are based on trust provided by payment systems. People purchase goods, book tickets, rent cars, etc. using international payment cards from Visa, MasterCard, etc. Merchants and cardholders trust payment cards because their systems take care of the financial risks involved. Autonomous banks work together under the umbrella of a payment card organisation, such as MasterCard, to provide a service [MC] that ultimately transfers money from the cardholder's account to the merchant's account across a number of financial service providers. Although it is assumed that everybody has adequate knowledge to observe the MasterCard Rules [MCRU], the system is also capable of detecting and handling fraud. Observing the correct rules, and having the power to enforce them, are key elements in creating an operation to deliver such a well trusted service. By combining the Generic AAA concepts with concepts from the card payment world, applied to e-Infrastructure context, we studied ways how authorization of resources can be arranged in a trusted way across multiple service providers. In the same way as for example a MasterCard credit card is perceived, such group is tasked to define an end-to-end service that is perceived and trusted by the user as if a single provider delivered it. By gaining better understanding of what is needed to create such trust and power in the MasterCard case, a framework was developed describing how autonomous service providers can act as a Service Provider Group. As each bank or financial service provider contributing to the MasterCard payment card service can be seen as a sovereign kingdom, this context resembled our multi-kingdom problem.

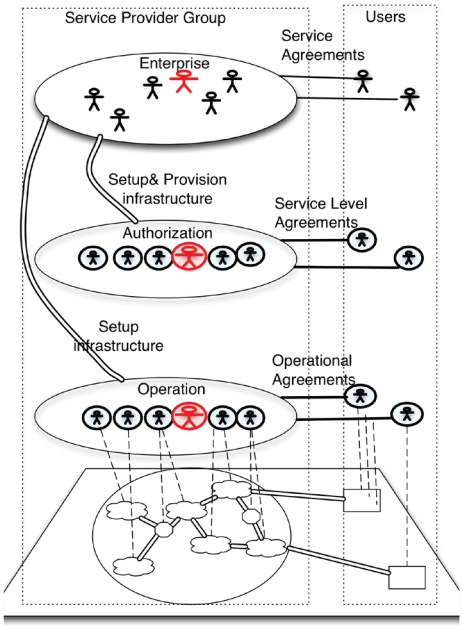


Fig. 2.5.7: The Service Provider Group concept as presented at the Chicago OGF meeting, Oct 2012. It shows that a SPG involves the consideration of three interacting layers.

Concepts of the research were presented and discussed at the OGF Network Services Interface Working Group (NSI-WG) [NSI] (fig 2.5.7), an Internet2 member meeting [INET2] and resulted in a publication as shown in table 2.5.3.

	Author Contributions	**Presented / published**
1	Trust Framework for Multi-Domain Authorization [TFMD]	Internet2 Spring Members meeting, Arlington, April 2012
2	Operating Framework for a Virtual Connection Network [NPG]	Presentation at OGF 36 NSI WG Chicago, Oct 2012.
3	The Service Provider Group Framework [GOM14]	Journal paper in FGCS

Table 2.5.3 Author contributions considering trust in the context of authorization.

Applying Generic AAA in e-Infrastructures

3

"I'm a great believer that any tool that enhances communication has profound effects in terms of how people can learn from each other, and how they can achieve the kind of freedoms that they're interested in."

Bill Gates (1955)
Co-founder Microsoft

3 Applying Generic AAA in e-Infrastructures

In section 2.2, we have introduced three different fundamental sequence models that can handle authorization requests. In section 2.3, we saw the Generic AAA architecture handle authorization requests by means of a Rule Based Engine driven by policies and Application Specific Modules handing the semantics of a request. We expected that our concepts would be applicable generically. In this chapter we will consider the applicability of these generic concepts by studying a number of example scenario's in the context of e-Infrastructures. Experiments, performed using these scenario's, will be described in chapter 4.

The application scenario's we will consider, have been placed in the context of (optical) networking, deployed as part of e-Infrastructures, by National Research and Education Networks (NRENs). Such placement was driven by a number of facts:

- The inherent research orientation of National Research and Education Networks, whereby multiple of such autonomous networks form an ideal environment to investigate our questions
- The inherent need for NRENs to collaborate in order to provide dedicated network connections at global scale. When data intensive e-Science applications require such network connections, NRENs can only provide them if they chain dedicated network connections to transport extreme volumes of traffic.
- The need for special dedicated (optical) network connections. The timely transport of extreme volumes of data at global scale would otherwise be too disruptive for the regular Internet.
- The need for policy driven allocation of network resources. As recently stated [ARNA], the network research world *needs to learn and deploy a policy management infrastructure that will allow them to seamlessly allocate resources and participate in multi-domain test-beds.*

Based on a series of publications (see chapter 8), this section will describe how the generic authorization concepts can be applied in a number of different ways in the above context. As such this chapter will address our second sub-question: ***What generic authorization concepts are expected to work best for classes of applications that uses multi-domain network resources?***

Considering the Generic AAA architecture, we first need to understand the relationships between the concept of one or more AAA servers being part of controlling a network when we consider a network as a collection of nodes interconnected by links. How do we see AAA servers controlling network nodes and links, in particular when a network is divided into multiple-domains? Can we recognize patterns in the way control can be organized? With what functions does a AAA server need to interact considering different kinds of network technologies and management functions. What scenarios do AAA servers need to support? How do we see these scenarios work? Does our Authorization Framework help?

Using the Authorization Framework, we must consider what sequence model(s) can be applied to scenarios that authorize network connections. What motivates further investigation of a particular model? If we can motivate a particular model, the question becomes: "How can such a model be implemented using a particular network technology? Is it possible to implement the concept in more than one way? In what multi-domain scenario does a recognized implementation fit, in particular considering the requirement that each domain wants to remain autonomous and, therefore, will resist sharing details? Using the Authorization Framework as basis, can we describe authorization interactions and access enforcement in more detail? If so, what protocols and technologies can implement these details?

The cases presented have in common that an authorization mechanism provides access to network connections with special qualities such as exclusive access, dedicated bandwidth, a particular route, no delay fluctuations, etc. We will consider cases as solution for single- and/or multi-domain scenarios. Single domain cases are typically used to first study principles that are also applicable to a multi-domain context. We studied authorization concepts by applying them at different functional layers in the network: IP packet level, control plane level and service level. Authorization messages can be carried and enforced at different network levels. The sections motivate why a combined push- and agent sequence, using a simple token to communicate decisions, are particularly useful for applications in the Grid & Networking e-Infrastructure context.

This chapter focusses on the elements that show how the concepts, defined in chapter 2, can be applied. Chapter 4 will focus on the demonstration of the applied concepts presented by the publications contained by this chapter.

3.1 The Agent sequence controlling a single domain path[3]

This section demonstrates the use of the agent model controlling lower level network equipment to provision a VLAN. It describes the use of the Generic AAA architecture implementing the Agent sequence. It will identify control models describing the role of a Generic AAA server can have in taking part in the control of network segments. The Generic AAA concepts were for the first time experimentally demonstrated at the iGrid 2002 conference in Amsterdam [DEF1]. The experimental part, demonstrating the applicability of the presented sequence, can be found in section 4.1.

For data intensive Grid applications, such as shown at iGrid 2002, users may require short- lived guaranteed high bandwidth connections. These types of connections, providing a certain Quality of Service (QoS), will need to be authorized and provisioned, often through multiple administrative domains. We present a case study of a Bandwidth

3 This section is based on publication:
"Authorization of QoS path based on Generic AAA", Leon Gommans, Cees de Laat, Bas van Oudenaarde, Arie Taal, iGrid2002 special issue, Future Generation Computer Systems, volume 19 issue 6, pp. 1009-1016 (2003).

on Demand service that provides as QoS path based on Generic Authentication, Authorization, Accounting (AAA), that represents a first step forward towards a multi-domain solution.

3.1.1 Introduction

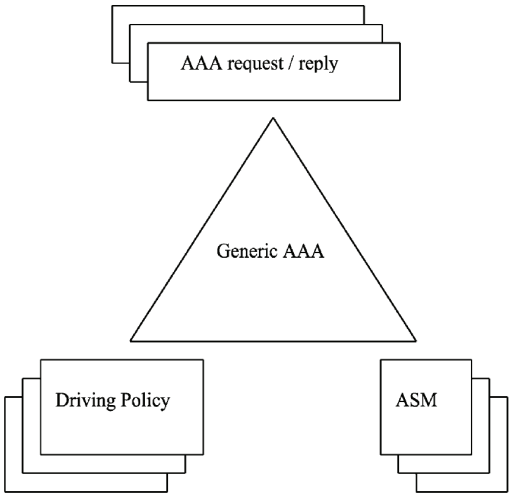


Fig 3.1.1 The different components of an AAA Server

Here we will discuss a middleware solution for the authorization of a Quality of Service (QoS) path with a focus on high bandwidth. QoS represents value and, when used in an on demand fashion, needs policy control to allow cost effective usage. QoS path creation was first introduced with the Resource ReSerVation Protocol (RSVP) [R2205]. During the QoS path creation routers exchange RSVP requests and have the possibility to pull a policy decision from a policy server using Common Open Policy Service (COPS) [R2748]. In our approach a network of AAA servers communicate Bandwidth on Demand (BoD) requests and use the agent sequence as defined in the Generic AAA framework [R2904]. An advantage of the agent sequence is that the underlying equipment does not have to carry as much intelligence as equipment using the pull sequence. Network equipment such as layer-3 routers, layer-2 switches or layer-1 cross-connects is capable of providing a QoS path with different levels of granularity. Because the agent sequence can control less intelligent equipment, our focus is on a switched-like architecture as the more appropriate solution to authorize a QoS path.

3.1.2 Generic AAA Architecture

An AAA server may be involved in handling one or more of three basic functions. The first function is Authorization of resource usage. Secondly, it verifies identities (Authentication). Finally, the AAA server may log key attributes of its functions so they can later be used for Accounting purposes. The architecture of a Generic AAA server is explained in [R2903]. We consider the authorization of a QoS path through multiple administrative domains. As for each administrative domain the authorization of its resources are implemented by an AAA server, more than one AAA server need to communicate by means of AAA requests in order to authorize a QoS path. A number of definitions exist for terms with respect to policies; see for example [R3060, LOBO]. In this section, we will introduce the term Driving Policy. In our applied model, an

AAA server fetches a Driving Policy when it receives an AAA request. For each type of AAA request, there exists a corresponding Driving Policy that instructs the AAA server how to deal with the request, for example a Driving Policy has to check the pre-conditions of the actions to be performed and how to deal with the post-conditions of these actions. We distinguish between specific actions and generic actions. Specific actions are performed by a so-called Application Specific Module (ASM), whereas generic actions are delegated to the generic part of an AAA server. The provisioning of a path within a single domain is typically performed by an ASM. Providing the date is an example of a generic action. The module that is responsible to execute a Driving Policy is the Rule Based Engine (RBE) and is contained in the Generic part of an AAA server. Fig. 3.1.1 shows the different components of an AAA server. The behaviour of the generic part of an AAA server is determined by the combination of Driving Policies, ASMs and AAA requests. This implies that behaviour can be easily adapted.

3.1.3 Authorization/Control models

In the context of Generic AAA, we consider a QoS path to be built out of two elementary QoS elements: network nodes (vertices) and network links (edges), where the relevant parameters of a network node or network link are under the control of an AAA server. This means that these parameters are governed by a set of policies residing in the Policy Repositories of the AAA servers in control.
In general a QoS path may span multiple administrative domains. The following terminology is introduced to facilitate the model discussion. We introduce the term QoS segment as the part of a QoS path that is under control of a single administrative domain. Each QoS element is authorized by an AAA server. Those elements of a QoS path belonging to a single AAA server are called a QoS component. The AAA server controlling QoS elements should also maintain the state of the assigned part of the QoS path.

In this way, the whole QoS path can be seen as a guaranteed end-to-end connection. Taking the simplest unidirectional QoS path between two nodes, three different control/authorization models can be distinguished: Individual Control, Partial Control, and Full Control.

3.1.3.1 Individual control model

Each element, network node or network link, is managed individually by an AAA server (Fig. 3.1.2). The AAA servers execute their own and independent set of policies. This model offers "Individual Control" of the simplest element. According to the used terminology, each QoS element is a QoS component.

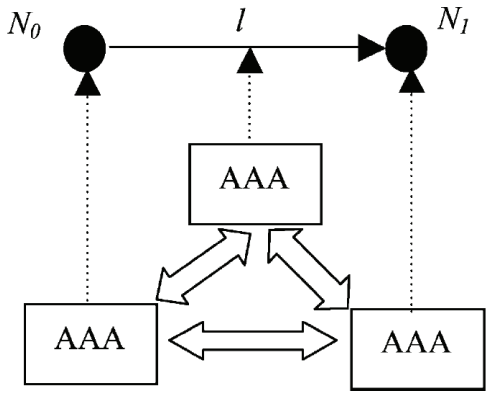


Fig 3.1.2 The Individual Control model.

3.1.3.2 Partial Control Model

A single set of policies controls the usage of both a network node and its outgoing network link (Fig. 3.1.3) or the set of policies controls a network node and its incoming network link (Fig. 3.1.4). This requires specific policies to describe the negotiations with the neighbouring element. Both figures consist of two components.

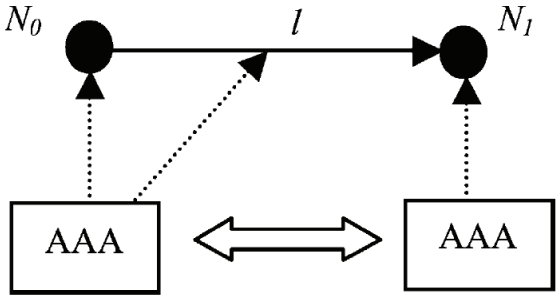


Fig 3.1.3 Partial Control model with an outbound connection.

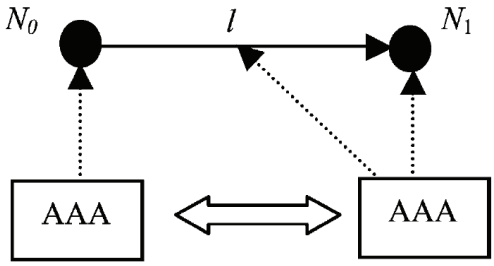


Fig 3.1.4 Partial Control model with an inbound connection.

3.1.3.3 Full control model

A single AAA server controls all three network elements of the simple QoS path, i.e. the QoS component is a segment (fig 3.1.5).

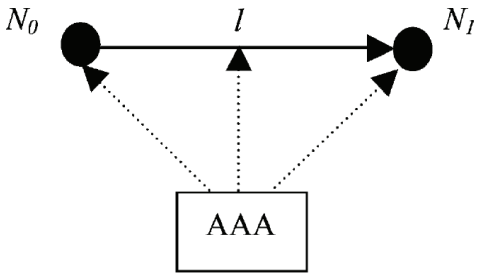


Fig 3.1.5 The Full Control model.

3.1.4 Authorized path discovery

To show how an authorized QoS path through multiple domains can be established, we make the following assumption. AAA servers, especially proxy AAA servers have an underlying mechanism that advertises the connections they can establish. Although not fully researched yet by our group, we expect to be able to re-use mechanisms such as BGP [R1771] (or proposed extensions to this protocol that is referred to as Optical BGP) to advertise reachability of networks within administrative domains. A purposely build BGP ASM will enable an AAA server to obtain a view on topology and discover which AAA servers should be contacted along the QoS path. Each AAA server will act as an agent or broker for its sub-domains.

As the start for the setup of an authorized QoS path between two nodes, $N_0 \in D_0$ and $N_n \in D_n$, one of the control models shown in Figs. 3.1.2 .. 3.1.5 can be selected. If D_0 and D_n are different administrative domains, the start situation will be the simplest QoS path according to the Individual Control model or according to the Partial Control model. If D_0 and D_n turn out to be the same

administrative domain, a simplest QoS path according to the Full Control model may also be an appropriate choice. In general multiple domains are in play, i.e. the network link of the start situation will be a logical network link instead of a physical network link. Authorization of a QoS path between $N_0 \in D_0$ and $N_n \in D_n$ is established through the following number of steps. Without loss of generality we take the Full Control model as the starting situation, with $D_0 = D_n$ and logical network link $\tilde{l}$ (see Fig. 3.1.6). N_0 and N_n are authorized by AAA_0, but for the logical network link $\tilde{l}$ a physical solution must be found. Therefore, AAA_0 will contact the AAA server that advertises a connection between N_0, N_n.

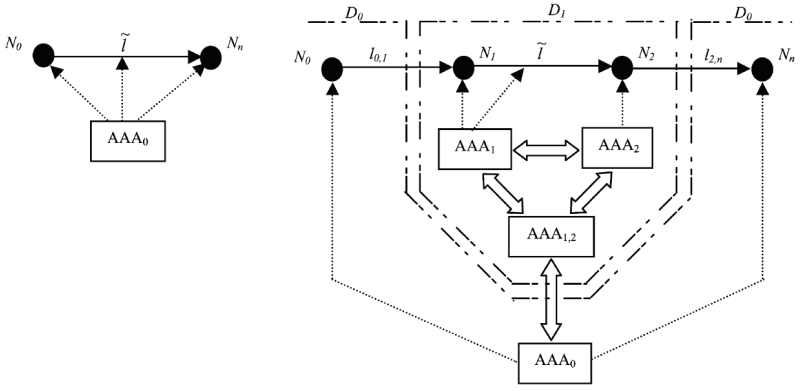


Fig. 3.1.6 The setup of a QoS Path starting from the Full Control model.

In Fig. 3.1.6 it is $AAA_{1,2}$ that advertised the required connection. Server $AAA_{1,2}$ acts as a proxy server for two other AAA servers: AAA_1 and AAA_2. AAA_0 and $AAA_{1,2}$ exchange requests to authorize the network links $l_{0,1}$ and $l_{2,n}$. Fig. 3.1.6 shows that for the authorization of $l_{0,1}$ $AAA_{1,2}$ resorts to AAA_1 and for the authorization of $l_{2,n}$ $AAA_{1,2}$ resorts to AAA_2. This whole process is started by a Driving Policy residing at AAA_0. According to the situation depicted in Fig. 3.1.6 this process of authorization will continue via AAA_1, as this AAA server still controls a logical network link. As soon as all logical network links of the QoS path are authorized, this process of authorization has come to an end.

The provisioning of the complete QoS path might be established using different approaches. One approach is to wait for provisioning until the all AAA servers in the path have approved. It is also possible to choose for an approach of immediate approval with rolling back the provisioning if one AAA server down a path refuses the request. More research is necessary to describe the role of policies in both of these approaches.

3.1.5 AAA server authorization interactions

For any QoS path a particular AAA server will be the root of a decision network (tree) needed to authorize usage of the QoS elements involved. All AAA server interactions are policy driven.

Policies may ask for a set of credentials or tokens to be present in the request that represents somebody else's authorization or delegation of authority. For example, these credentials or tokens may have been issued by an advance reservation system as a proof of a reservation.
Policies may involve parties in the decision that may or may not be hidden from the requesting party. For instance, a budget or payment authorization may be requested. Such checks may additionally involve parties unknown to the requestor. Next to the content of an AAA request, that may need to carry a token or credential that represents an authorization, also the integrity, peer authenticity and sometimes confidentiality of a request must be ensured. Here we rely on existing security mechanisms such as provided through the GSS-API [R2743] that offers support for these aspects. More specifically, toolkits such as Globus's Grid Security Infrastructure does also rely on the GSS-API. Some level of interoperability is offered when the GSS-API is deployed with Public Key certificates issued by a GRID Certification Authority. The VOMS project [INFN], within the EU DataGRID [DAGR] and DataTAG [DATA, OUDE] project researches the definition of roles within Grid environments. Through collaboration with the VOMS project we plan to integrate the concept of roles. A user role may define that a particular user is authorized to allocate bandwidth.

A case study based on the described way of using Generic AAA servers within the context of scientific Grid users will be presented in section 4.1.

3.2 Token based authorization[4]

This section presents a theoretical study arguing the use of a special device and tokens to manage access to a network path across multiple domains. The study of such special device leads to the development of the token-based switch that will be described in section 3.3.

Authentication, Authorization and Accounting (AAA) mechanisms have an increasingly versatile role when performing access control on various types of network resources. Emerging data intensive grid applications generate new network requirements. These requirements call for (pre-) allocate-able data transport facilities serving specific user communities. The network specific requirements are characterized by a very limited need for connectivity, usage during specific periods and in many cases the need to span large distances at maximum available speed. This should all be possible at the lowest achievable cost involving different owners of the involved networks. Solutions are researched in the area of connection oriented networking using relative cheap, lower layer switches. This section presents a novel usage of an existing model to grant access to connection oriented network resources based on an authorization message sequence involving a token. The presented approach makes use of specialized monitor hardware emerging in high capacity low-layer switches.

4 This section is based on publication:
"Token-based authorization of Connection Oriented Network resources", Leon Gommans, Franco Travostino, John Vollbrecht, Cees de Laat, Robert Meijer, GRIDNETS conference proceedings, Oct 2004.

3.2.1 Authorization sequence models within traditional networking

We have seen three different Authorization Framework sequence models in section 2.2. The pull- and agent models for authorization sequences are quite common within networking. In this section we will give some examples as they are being used for a number of different applications in the area of access control and QoS networking.

3.2.1.1 The pull sequence

The pull sequence, as described in section 2.2.2, is typically used whenever a user tries to gain access via some device that is offering service and is also capable of enforcing access. This device could be at the ingress of a network. As an example of such an enforcement device we describe a Network Access Server (NAS). A NAS (see 2.1.2) is used to recognize a user dialling into an Internet Service Provider using a modem. Whenever the NAS receives a call on one of its phone lines (fig 2.1.5), it will pick up the line and connect a modem. After the modems synchronize and line protocol has been established, an authentication protocol such as (CHAP [R1994] or PAP [R1172]) will exchange user-identifying information with the NAS. The NAS is configured to contact an AAA server in order to send it the user related information using a protocol such as RADIUS [R2865]. Multiple NASs may contact the same AAA server. One AAA server will typically serve a single domain of users. A so-called User Home Organization (UHO) could operate the AAA server independently. A university could administer such a server on behalf of their student community. If permitted, each service providers NAS could in theory access a universities AAA server. As this requires every NAS to know about an additional university, this model does not scale very well. One approach to the solution is to use proxy chaining as described in RFC 2607 [R2607]. Another example would be RSVP [R2205] as briefly mentioned in section 3.1.1. Here routers pull authorization decisions to treat certain traffic flows differently from a policy server.

3.2.1.2 The agent sequence

The agent model is typically used whenever a user or a service does not have the knowledge and/or relationships to obtain a particular authorization. Agents abstract the underlying complexity and offer the authorization service in a simplified and/or easier to use way towards the user of the agent. Agents typically advertise themselves by a well-known mechanism. An example of an agent sequence is used in a bandwidth broker [QBON] scenario (fig 2.5.2) within networks that deploy differentiated services [R2474]. A broker could keep track of the available capacity inside a domain. A chain of domains will implement a certain capacity between a source and destination. Bandwidth brokers will provision routers inside a domain with the proper queue parameters to implement a certain diffserv model. Bandwidth Brokers will communicate with neighbouring Bandwidth Brokers to ensure a certain bandwidth is available when traffic traverses the underlying domains. A user will typically communicate with a Bandwidth Broker at

the source domain. This bandwidth broker will effectively authorize traffic if conditions checked with neighbouring Bandwidth Brokers so permit.

3.2.2 The use of the push model at the lower network layers

The push model is used at the application layer predominantly. Many examples are found where signed tokens or certificates enable applications or systems to recognize users, user privileges in order to provide access control functions. Access is gained after a user first obtains a token from the AAA server and subsequently presents this token to the Service Equipment. A certificate, in comparison with a token, suggests a particular format. A token is a more general type of trusted, cryptographically protected proof of authorization with less strict issue and usage policies. A token cryptographically binds attributes to an issuing attribute authority, not to a user. Therefore, a token could be acquired and subsequently used anonymously. In our case the AAA server acts as a kind of attribute authority that issues something like an Attribute Certificate. As the terms Attribute Certificate and Attribute Authority have already been defined in RFC3281 [R3281], we refer to our AAA server issued, cryptographically bound set of attributes more generally as a token, without assuming any particular format like X.509 [X509]. Within this context we assume a token to be a set of attributes that is cryptographically bound (signed) by the AAA server acting as authority. Signing proofs the authenticity and integrity of a token. It neither prevents duplication nor ensures confidentiality. A token could be used in several ways: It could for example periodically handed in a secure fashion to the network or it could be used as key material with some message security method that is used along with every message (eg. some encryption or signing method).

3.2.2.1 Content Monitoring and Action Device

Lower layer network transport functions work in a connection-oriented way. These functions do typically not recognize tokens inside signalling messages or as part of the data stream. In this document we intend to drive network provision functions based on its recognition. We make the assumption that a token can be recognized even at the lowest network layer possible. This implies recognition at layer 1 or 2 switches. As part of elaborate network intelligence gathering capabilities, switch manufacturers develop hardware functions that are capable of recognizing network content at wire speed. This hardware includes programming capabilities that can subsequently trigger and execute actions. We will refer to this kind of device as a Content Monitor and Action Device (CMAD). Although one can imagine that such device could be implemented on photon-based switches, engineering efforts currently focus on switches that perform their functions using electrons. Assuming the existence of such device, we now consider possible application of a CMAD with regards to the authorization of special network resources.

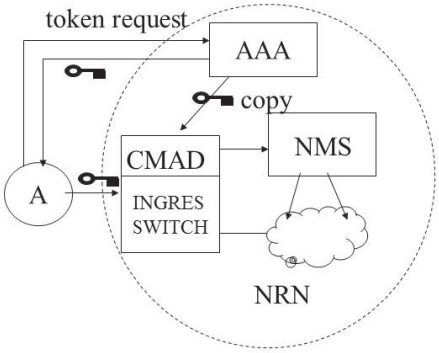


Fig 3.2.1: A possible position of the Content Monitoring and Action Device

Fig. 3.2.1 shows a possible position of a CMAD as part of an ingress switch at the edge of a network. The token, represented by the key symbol, is handed to user A as a result of the push sequence. The token or a derivate of this token is subsequently recognized by the CMAD when it is handed to the network via the ingress switch. When a token is handed, some method must be used to avoid replay attacks. If the token is used the AAA server may already have provided a copy of the token to the CMAD as to ensure fast recognition. The CMAD could signal the network's Network Management System (NMS) to implement the desired connection inside the network cloud of the NREN (short NRN). There are many more sequences and software components such as a resource manager that will play a role in the implementation of such a network. However, for simplicity reasons, those sequences and components have been omitted in Fig. 3.2.1.

3.2.2.2 Rationale

Let us first consider the rationale behind a token-based network. There are some fundamental differences between the push model and the pull or agent model. First, the push model allows a separation in time between the issuing of an authorization token and the usage of the token. The pull and agent model assume an authorization to be taken at the time of sending a request. The push model assumes that the issued authorization token will be used at a later point in time. Second, if the issuing policy so permits, the entity requesting the authorization token may be different from the entity using the token. Third, a token from a central authority may autonomously be recognized by a number of different services or service domains without the need for further communication. This approach fits well in a centralized authorization model where each domain is still able to take autonomous decisions on the presented token. For example, a policy may decide that a token may only be presented at selected ports. As tokens can be carried along with an application, a token-based approach could for example simplify operational aspects when considering pre-allocation requirements in grid environments.

3.2.2.3 Goals

The goals of using a push model can be summarized as:

1. Allow a token to represent a pre-allocated network connection that spans multiple domains for a specified amount of time between two locations.

2. Allow the creation of various token distribution models where a token can be requested and subsequently be handled by (a chain of-) organizations that act on behalf of the ultimate user.
3. Allow fast creation of a connection based on pre-configured information that is referred to by the token. This information is already established at token setup-time.

3.2.3 Example use case

3.2.3.1 Definition of the network

Consider a federation of NRNs that are served for their interconnectivity by an organization that involves a number of Global Network Carrier (GNCs). The Federative Network Organization (FNO) is responsible for maintaining contracts with individual NRNs and with GNC. The FNO could operate its own carrier network but for simplicity reasons the FNO and GNCs are considered separate functional entities. The FNO is also responsible for the financial clearing and settlement between its members.

3.2.3.2 Maximum bandwidth connection

Federation members offer both best effort and special maximum bandwidth connections between a set of well-defined and static source and destination locations that serve a certain community. Maximum bandwidth connections do not experience any packet loss either from sharing network resources with other users or by any rate-limiting mechanism. A maximum bandwidth connection allows usage of aggressive protocols intended to maximize the bandwidth utilization. We assume that grid applications at the endpoints require either pre-allocated or on-demand maximum bandwidth connections. The rationale behind this assumption is described in [DLA3].

3.2.3.3 Using and obtaining a token

At the time the application needs the pre-allocated bandwidth, the user will insert a token in the network. This can be done either as a signal along with the data stream (in-band signalling), or the token can be handed to a special signalling interface (out-band signalling). In order to obtain tokens, the user or a representative user organization contacts the federation AAA server with an appropriate request. The information in the request is based on the advertised availability of specific connection that is available to the public or to a certain community. A request may be made for a specific period of time starting now or at some future point in time. After the request is received the FNO will contact the involved resource domains to find out if a reservation is still possible. If all involved resource managers of the domains agree, the FNO will subsequently

allocate all resources both inside the NRN domains and also allocate an interconnecting link from the GNC domain. In our example the FNO is responsible for the resource management of the GNC links. A GNC could also have a separate function for its resource management. Only if all underlying resources are available, the final allocation will be registered. The FNO is considered the authority that is contractually allowed to allocate the advertised network resources with NRNs and the GNC on behalf of a requesting NRN. A requesting NRN is contractually allowed to make further refinements or subdivisions to the offered service on behalf of its users. Users are contractually allowed to do the same. The GNC is in this sense the "manufacturer" of the bandwidth and the FNO and NRN's are considered "distributors" of bandwidth. NRNs typically lease a set of connections for large periods of time with a GNC. The FNO is allowed to subdivide this bandwidth between NRNs. The GNC is therefore in our example not concerned with the actual usage.

3.2.3.4 A token request and service ID

A source NRN is typically the requestor, but in theory this would not be a requirement. One can imagine applications need bandwidth between A and B and subsequently between B and C. Source NRN A is allowed to make reservations for bandwidth between B and C. A NRN does not need to know to whom this bandwidth has been allocated. A NRN will trust the FNO for this. The involved parties will however want to know about a unique service ID that the FNO has assigned to the allocated timeframe. The service ID may also have sub-ID's that will point at a particular time-slot within an allocated timeframe. Based on this service information, the FNO will generate a number of tokens. Each token will point at an individual time-slot and reference to a specific link within the allocated timeframe. The tokens receive a cryptographic proof of authenticity from the FNO that each NRN will recognize. A token will also point at a reservation for a specific link. When a token is received, it is the responsibility of the NRN to map this pointer to a particular link.

3.2.3.5 Receiving a token

Before receiving a token, a domain could have already prepared all the necessary control parameters for each individual switch component in advance. There will be some optimal time period for doing these preparations before a token is expected. This time will for example depend on the likelihood of failure between the preparation and the actual usage of the link. One may also determine an earliest time to receive a token and assign significance to the token. Tokens received before such time should be ignored. As shown already in fig. 3.2.2, once the token is recognized, a domain specific network management system could provision all involved switches with the correct configuration information as to establish a connection. A NRN should also maintain a resource management system to be able to administer network allocations and this system should also be able to generate the necessary element configuration information so that

a network path could be activated very rapidly. The FNO, who in our case does the resource management for the GNC links, should allocate an inter-NRN link and should identify the selected link to the corresponding NRNs such that the correct inter-NRN link is used.

3.2.3.6 Distribution of tokens

If the NRN, on behalf of the user community, obtains a set of tokens from the FNO, the NRN has the right to distribute these tokens according to its own set of rules to one (or more) of its customers. The NRN however cannot further subdivide a token, as this would mean a security breach for the token. The NRN should distribute tokens by some secure means to its user, ensuring confidentiality and avoiding unwanted duplication of tokens. Intermediate parties could hand copies of the same token to more than one of its users. Such action would allow a connection to be shared. In our simple use case, there is no binding of the token to the identity of a certain user. Therefore, anybody in the possession of the token is able to use the token. The token must however be used from the source domain. A policy may determine if the token is only accepted on certain ports. Duplicate token usage from different ports may also be denied, based on a policy, to prevent sharing.

3.2.4 Token Requirements

The described use case raises the question on the kind of information that should be contained within the token and how a token should be secured. It is not the objective of this section to cover this topic in detail as much research is left to be carried out. Here is merely a group of requirements towards such a token. At token must contain:

- Some proof of authenticity that is recognized by multiple service domains.
- The validity period (start- and end-time) of the token.
- An optional (encrypted) list of service domains in which the token is valid.
- A unique reference number that can be used for accounting purposes and allow providers to collect this information.
- A reference to a pre-established service instance. This service instance references all necessary information to instantiate a connection.

Towards usage the following requirements can be given:

- Tokens can be send once or multiple times. If received more then once, the additional usage re-asserts the usage right.
- If the service fails within the designated contract period, the token should be kept as reference.
- Token lifetime should be limited and granular enough to support application demand.

- Tokens should be allowed to be send some prior time to the validity of the token in order to maintain uninterrupted service.
- If no token is received, the connection between two domains may be based on best effort service.
- A token will always be received on a network ingress point that initially always connects to a best effort routed or VPN service. The token will cause the network to switch to a maximum bandwidth service.
- If the validity of the token is expired and no new token is received, the network will revert automatically to the best effort service.

3.2.5 Example network

Consider the network of fig. 3.2.2, where end stations A and B want to communicate via a maximum bandwidth connection. A is connected to NRN-X via a single Gigabit Ethernet connection and station B is connected to NRN-Y via a similar connection. NRN-X and NRN-Y are connected with multiple connections via a GNC.

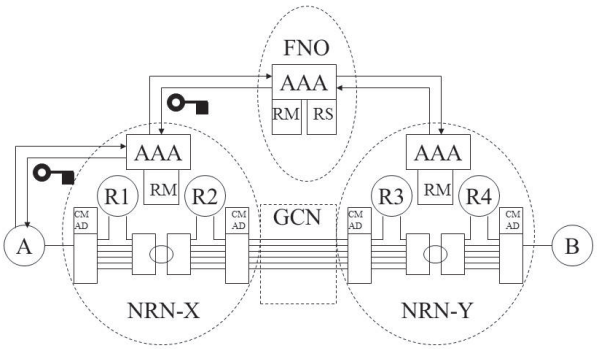


Fig 3.2.2 Example Architecture featuring two NRNs interconnected through a Global Carrier Network where the Federative Network Organization is responsible for the Resource Management (RM) of the GNC.

Each NRN has a core transport network (e.g. a SDH ring). Switches at the ingress (client) and egress (GNC) side of the network determine the kind of connectivity offered. By default, the network offers a best effort service. The ingress/egress switches are also capable of accessing the core network directly. By default station A is connected via a VLAN to the router R1 of NRN-X. The NRNs transport infrastructure will interconnect R1 and R2. The egress switch of NRN-X and ingress switch of NRN-Y make sure that the two border routers R2 and R3 of each NRN are interconnected. The transport infrastructure of NRN-Y will interconnect R3 and R4 of NRN-Y and the ingress switch of B will connect R4 to B. Each ingress and egress switch of an NRN has a CMAD to monitor every port of the network for incoming tokens.
The architecture of an AAA server is subject of research by the AAA Architecture Research Group [AAAARG]. RFC2903 [R2903] describes such an architecture. An important aspect of an

AAA server is the fact that it uses a driving policy to consider policy conditions and to take subsequent policy actions. The issuing of tokens is the responsibility of the FNO's AAA server. Upon receiving a request message from an NRN, the driving policy causes the AAA server to first contact a routing service (RS) that is capable of identifying one or more routes involving one or more (NRNs). Each NRN will advertise all available routes to this RS. Then the driving policy will contact all the involved resource managers (RM) to determine if a particular connection is available at the desired time.
First it will check with its own resource manager. The FNO RM overlooks the usage of the GNC links between the NRNs. If a connection between the two NRNs is available, the driving policy will contact both NRN RMs of the source and destination domain via the NRNs AAA server. It will request if a transport connection is available from the egress switch connecting to the GNC to the ingress switch of A and B at the given time with the requested capacity. Each domain Resource Manager will provide this data, but a policy at the AAA server may still permit or deny a particular use. If both domains answer are positive, the FNO's AAA server will generate and sign the requested amount of tokens and send them to the AAA server of the requesting domain. It will also contact the RMs make an allocation of the appropriate resources with the proper ID and will send a copy of the signed token, which can be compared with the token received from the user.

The requesting NRN's can store these tokens and issue them to a NRN user upon a request or the NRN can act on an user request directly depending on the model. The NRN could already have known the need based on a contract with the user or the user could have ad-hoc demands. If the user receives one or more tokens from the NRN, it may further distribute the tokens to its applications, which will then insert the token into the network via an in- or out- of band method.

3.2.6 Network Considerations

The pictured method of creating a Layer 1 or Layer 2 bypass connection after regular communication between station A and B goes via a routed connection is fairly trivial if one takes care of the proper end station configuration and the proper support at the router network. Firstly the end stations must think that they are always talking to each other via a L2 network. This means that the default gateway of a station must be set to its own address, causing the station to ARP [R826] for every destination IP address. The routed network must therefore support proxy ARP, which is typically enabled by default. If the network bypass connection is created and put into effect, the end station ARP caches need to be flushed to re-learn the MAC – IP address association. This also needs to be repeated whenever the by-pass connection reverts.

3.2.7 Conclusions

The token model allows control of network resources involving different domains. The token represents the right to use a pre-established network connection at a specific time domain. The

generation of a token for a future usage right, allows the trading of the token between the time a token is generated and the time the token is used. This fact allows for various trading models. One organization could order a bulk of tokens and resell and/or distribute the usage rights. As the token is not bound to a particular user, the user is responsible for maintaining the security of the token. Modern hardware devices present in switches can be programmed to recognize tokens in the data stream. This may omit complex signalling interfaces. Switches at the ingress points or at a central point of a network could be holding such monitor and action device. The user must insert a valid token at defined intervals in the data-stream, keeping the network by-pass connection alive. This user may or may not be the same user as the previous user inserting a token. Policies at individual domains may restrict the usage of a token. We have not tried to describe a detailed solution as this is for future study.

3.3 Token sequence authorizing network level access[5]

This section will motivate that a meaningless token, referencing a service instance (see requirements of section 3.2.4), is a viable mechanism that can be used at network level. This mechanism allows enforcement of an authorization allowing access to a network path supporting guaranteed bandwidth that can be offered by a "Lamda Grid".

In order to provide cost effective transport services for highly demanding data-intensive grid applications, National Research Networks (NRNs) are considering additional types of access to their network infrastructures. Next to traditional IP access, NRNs like to provide automated, grid application driven access to their underlying connection-oriented network infrastructure. This combination is called hybrid networking. Recently, both NRNs and grid communities started to acquire their own global optical network connections. A Lambda Grid was created when these organizations decided to interconnect these links via interconnection points such as Starlight and Netherlight. Apart from supporting scientific applications, the Lambda Grid allows network research. Within this context, we will present a novel token-based path selection mechanism that will enable authorized access to Lambda Grid links. The token-based approach allows temporal separation of the path authorization process from obtaining access to a Lambda Grid link. The path authorization process may involve many parties and complex, time consuming decisions whereas path access requires a fast real-time implementation. We will describe the application of the token-based approach for an interconnection point between a Hybrid Network and a Lambda Grid.

3.3.1 Introduction

Recent regulatory changes enabled NRNs and scientific organizations to acquire their own global transport connections. Such connections are used to by-pass the existing Internet allowing scientific applications to run at lower cost. Collaborative efforts merged these connections into

---

5 This section is based on the first part of publication:
"Token Based path authorization at Interconnection Points between Hybrid Networks and a Lambda Grid", Leon Gommans, Cees de Laat, Robert Meijer, IEEE GRIDNETS2005 proceedings, ISBN 0-7803-9277-9.

an infrastructure called a Lambda Grid. Allowing owners of Lambda Grid links to authorize specific user applications to choose a more economic route is an evolution of the Internet that is addressed. We will present an in-packet token-based approach that can be integrated with an automated network path setup-, reservation- and authorization system. In order to position the approach, the paper will first explain the structure of the current Internet. NRNs can make Lambda Grid links accessible via their hybrid networks. We will therefore need to describe the basic concepts behind hybrid networks and show how hybrid networks interconnect. The presented approach performs authorized path selection at interconnection points between a hybrid network and the Lambda Grid. Grid compliant implementations of network path selection- and control functions typically involve web services mechanisms. Recent availability of cryptographic network hardware and disappointing experiences with the performance of web services has led our research to a novel, token-based authorization approach. A high performance switch equipped with the aforementioned cryptographic hardware acts as a real time path selection mechanism, using a token-key to recognize tokens inside IP packets. Before it issues a token-key to a network user, a web services based authorization process could involve several stakeholders and take complex decisions. Next to technological aspects, economic and regulatory factors are used to introduce some of the evolutionary aspects of the Internet. We will assume some conceptual knowledge behind authorization such as described in RFC2904 [R2904] and GFD.38 [GFD38] and presented in chapter 2.

3.3.2 Internet Interconnection principles

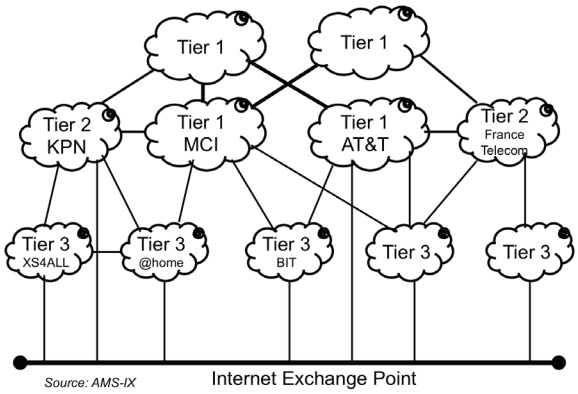


Fig 3.3.1. The hierarchical Internet structure.

The evolution of the telephone network in the US created a hierarchical network structure in which the FCC regulated fair competition amongst long distance carriers. In 1982, an anti-trust case [CA82] was settled where AT&T was forced to split off its long-distance services and divest its 22 local telephone companies into 7 separate regional operating companies. Legal cases, such as the AT&T case, did influence the structure of the Internet since the Internet backbones were all owned by Telco's. National regulatory bodies such as the FCC closely watched the telecommunications industry structure as to promote competition. This evolved the Internet into a three-tier structure as shown in fig. 3.3.1. Within this structure, Tier-1 carrier networks such as AT&T and MCI provide both long distance transport and transit services. Tier-2 networks provide regional transport and transit services. Tier-3 networks are national Internet Service Provider (ISP)

networks. A Tier-3 network may obtain transit and transport from a Tier-1 or -2 network. For simplicity reasons, we refer to the combined set of Tier-1 and Tier-2 networks as the Transit Network. A Transit Network, by definition, provides connectivity to the entire Internet.

If an ISP discovers that it exchanges a significant amount of traffic with another ISP, both ISPs may find it more economical to create a direct link between each other [NORT]. Traffic will then be routed via this link, by-passing the Transit Network as shown in fig 3.3.2. Policies inside routers that border other networks, will determine the desired routes of the traffic. Border routers use a protocol called the Border Gateway Protocol [R1771] to communicate available routes. When a group of ISPs see similar needs, ISPs may decide to form an Internet Exchange Point (IXP). At this point ISPs can peer with each other. The bottom bar in fig. 3.3.1 shows an Ethernet segment that forms an exchange. It connects Tier-1, 2 and 3 networks via a common network. For simplicity reasons, we again consider the collection of Tier-1 and Tier-2 networks as the Transit Network.

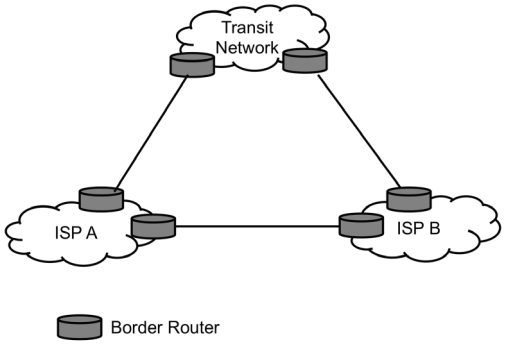


Fig 3.3.2. Direct ISP Peering

This simplification yields fig. 3.3.3. According to fig. 3.3.1 Tier-1 and -2 networks, also connect to an exchange. Fig. 3.3.3 is therefore not entirely accurate. Norton [NORT], however, uses a similar model as fig. 3.3.3 when explaining the business case for peering. We will therefore continue to use the simplified Norton model.

The admission policy to the Amsterdam Internet Exchange (AMS-IX) is called open: any legal entity with a registered Autonomous System Number [R1930] is allowed to become a member.

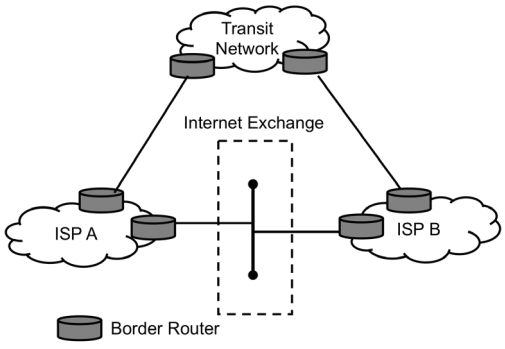


Fig 3.3.3. ISP peering via an Internet exchange point

Other exchanges may not reveal their admission policy or pose restrictions on data-flows.

However, in our paper we assume that an exchange has an open admission policy and does not impose any restriction. This means that an exchange should allow any member to create a peering relationship with another member autonomously.

When peering connection-oriented hybrid network links, one must observe similar requirements. Let us now consider hybrid networks in more detail.

3.3.3 Hybrid Networks

In traditional IP-networks, the collection of high-capacity links that interconnect routers is called the low-level infrastructure. It uses layer 2 or below technologies to transport data. Regulatory changes, such as the 1996 Telecommunications Act in the US, enabled companies to acquire telecommunication services competitively, including optical fiber links. Tier-1, -2 and -3 networks and even end-users could now start to own fiber-optic links. This enabled IP network operators to offer cheaper lower-level point-to-point transport services, thereby creating hybrid networks. Hybrid Tier-1 or Tier-2 networks could offer both IP transit services and point-to-point transport services. A hybrid ISP could provide point-to-point transport or transport from one point to an upstream network. Alternatively, it could peer with another ISPs. This peering can either be direct, using a peering link, or peering can be done via an exchange point.
As seen in section 3.3.2, traditional exchanges use a common Ethernet where flows are determined by BGP peering policies between corresponding networks. An exchange, supporting connection-oriented flows from hybrid networks, must perform a switching function to select a certain path. Considering the admission policy requirement mentioned at the end of the previous section, the path selection should only be authorized by policy-based decisions performed amongst the peering entities.

3.3.4 Hybrid Internet Service Provider peering

We saw that peering between traditional ISPs is done for economic reasons. We assume the same for hybrid networks based on the following rationale:

1. Lower layer connection-oriented switching between specific data-intensive sites is cheaper then connection-less routing [DLA3].
2. There is an economic break-even point between peering and transit when transporting increasing volumes of data [NORT].

If there is a choice between connectionless and connection-oriented peering, the above rationale will favour connection-oriented peering for high-volume data transfers.
Tier-1 networks offer both transport and transit at global scale. Transit, for connection-oriented flows, must be translated into the ability of a hybrid network to connect to multiple destinations on demand. This on-demand capability can be implemented by offering a control-plane interface into the network. Several web services based control-plane interface implementations are being researched in projects such as UCLP [UCLP] from Canarie and the Generic AAA project from University of Amsterdam. Considering the involved economics, a hybrid ISP must decide, if it connects to a Tier-1 or -2 network or if it peers with another ISP via an exchange. In any case, the involved networks must allow some form of control signalling and some mechanism must exist to select and setup an end-to-end connection. When peering, the exchange must also allow control signalling to select a path that is part of an end-to-end connection. The following consideration, at least from a scientific grid viewpoint, makes path selection for hybrid networks

more complex. We observe in the scientific grid world that optical global transport connections are scarce resources typically owned by organizations, who are either a NRN or a stakeholder in a grid community. Global optical connections are mostly acquired for longer periods and are therefore typically statically provisioned. As more NRNs and scientific communities started to own such connections, the Global Lambda Integrated Facility [GLIF] was formed in 2003. When created, an important aim of the GLIF was the development of functions and services allowing flexible use of what is called a Lambda Grid. Within this network, global transport connections terminate at special interconnection points such as Netherlight and Starlight. An Inter-Connection Point (ICP) is a general term for the technical facility that allows termination of network connections with the intent to interconnect. The GLIF essentially creates a network of ICPs, allowing scientific applications to peer at a global scale. The current drawing of the GLIF shows that for example CERN and SURFnet own global connections. CERN, when moving terabyte-scale data files on behalf its LHC experiments, might want to restrict other access. SURFnet may decide to allow any application provided it serves a scientific goal. A hybrid ISP network may or may not sit in between a GLIF interconnection point and the user application. NRNs, acting as ISP or Tier-2 network, position themselves to play a role between Grid applications and the GLIF. The network of ICPs and global links, where link owners plays a distinct role in admitting data-flows, creates a new networking situation. We assume that the transit network is the IP network by-passed by the connection-oriented network. Instead of an exchange, a network of ICPs creates the by-pass. A simplification of this situation is shown below.

Fig. 3.3.4 shows that user applications U and V have a choice to peer directly via ICP-A and ICP-B or connect via hybrid ISP-A and ISP-B that connects to ICP-A and ICP-B. Links, owned by one or more stakeholders, interconnect ICP-A and -B. When considering the GLIF, ICP A could be Netherlight in Amsterdam and ICP B could be Starlight in Chicago. Stakeholders, such as CERN or SURFnet, may want to authorize applications U and V to peer via its Lambda Grid link. Being connection-oriented, ICP-A and ICP-B both enforce access to a Lambda Grid link.

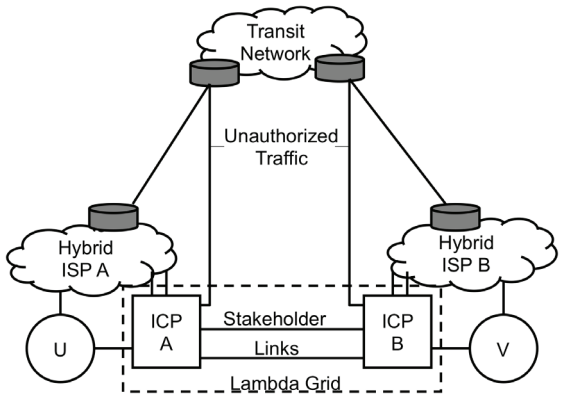


Fig 3.3.4. Peering via a Lambda Grid.

ICPs and hybrid networks need to signal each other to create an end-to-end path. In his description of exchange models, Dijkstra [DIJK] recognizes that policy based authorization mechanisms can be used to signal and control optical exchanges. Dijkstra also recognizes that peers are responsible to control their own network interfaces on an exchange. We can extend this observation with the recognition that Lambda Grid link owners have similar responsibilities. We conducted experiments to perform

end-to-end path authorization signalling involving multiple autonomous networks. During these experiments, network control interfaces were cast as a web service. Let us now consider some results of these experiments and explain why the token-based authorization approach is helpful.

3.3.5 Web Service performance and Token Based Authorization

Implementations of modern low-level infrastructures offer elaborate management and control capabilities by means of an Operations Support System (OSS). Examples studied were Nortel's DRAC [DRAC, TRAD] and Alcatel's 1355 BonD [BOND]. In general, an OSS performs management, inventory, engineering, planning, and repair functions for communications service providers. These functions represent a single administrative domain. An OSS creates specific connections on behalf of the administrative owner of a network domain. The owner, however, does typically not allow an outside entity to control the provisioning of its connections. We developed an authorization system based on the principles of the Generic AAA Architecture [R2903]. In several experiments using this system, an AAA server acts for clients as a proxy or broker of an OSS. Oudenaarde [OUDE, GOM61] describes experiments that shows how AAA servers can set up end-to-end path across a set of connection-oriented networks. One conclusion of the experiments expressed concern about the performance of a web services based control interface implementation. Experience from these experiments also showed that the ease of combining a set of services into a distributed application is the main value of web services. These conclusions create a dilemma when considering authorization mechanisms that involve web services interfaces.
Our research into this issue, and the recent emerge of network switching technology offering cryptographic functions, made us consider the RFC2904 [R2904] push sequence. With the push (or token) sequence, the user makes a request to the AAA server. The AAA server applies a set of policy conditions to authorize the request. Instead of sending commands to the service, the policy action returns a signed list of attributes called a token. The token could be used anytime or at an agreed time. When presented to the service, the authenticity of the token proves to the service that the AAA server has issued the token and the service is expected to act accordingly. This model effectively decouples the taking of a decision from the usage of the decision. A token can be assigned a validity time and can be kept, stored and maybe handed over to another user. If a token contains a token-key, the token-key could be used to generate a new token. These tokens must be securely communicated as signing does not provide confidentiality. If the AAA server acts as a proxy to a resource allocation manager, and if bound to corresponding attributes, a token allows precise resource management and pre-allocation. A token can be requested well ahead of time. These features are expected to help us with solving the timing issues when collecting authorizations from different places using web service interfaces. Once the authorizations are collected, a token could provide real-time access to a network link.

Based on the above recognitions, section 4.3 will describe experiments performed using and Intel IXDP 2850 platform as token based switch that was presented during the iGrid 2005 conference. It is a continuation of this section and will also include the conclusion.

3.4 Token sequence authorization applied to network cases[6]

The publication presented in this section was written to provide an overview of the concepts studied so far, which lead to demonstrations at SuperComputing 2006 and 2007.

This section highlights the concepts and results of our research, leading to demonstrations during the period 2005–2007 to develop a flexible and simple access control model, and corresponding support tools to provision multi-domain optical network resources on demand .We introduce the general network resources provisioning model that extends the Generic AAA Authorisation sequences for multi-domain scenarios, and explain how token based access control and policy enforcement can be used during the provisioned resource access. To build a solid conceptual foundation for the proposed token, based access control, the paper revisits existing token definition and proposes a new definition in the context of our research. We subsequently show the use of tokens during different stages of the lightpath provisioning process. The paper identifies and describes two major scenarios in multi-domain lightpath provisioning: the chain and tree approaches. The proposed token concept allows a simple combination of access control enforcement at different networking layers: the packet layer, the path layer, and the service layer. Section 4.5 will briefly describe a few demonstrations that proves the proposed concepts and illustrates its acceptance by a wider networking community.

3.4.1 Introduction

Modern high performance distributed applications, dealing with high volumes of data, increasingly require dedicated high-speed optical network connections that are provisioned in an on-demand fashion. This type of resource is commonly referred to as a lightpath [JWU]. Projects, such as OptIPuter [OPTI], envisage a LambdaGrid, where lightpaths are tightly coupled with computational resources. A LambdaGrid coordinates dynamic provisioning of end-to-end circuits using Grid concepts. On the other hand, large Grid projects such as the LHC Computing Grid [LHCG] use their own dedicated network infrastructure, designed to handle the required data volumes without being tightly coupled to computational resources. In our paper, we will not target such applications, but consider data intensive applications that are expected to benefit from the ability of a network to dynamically allocate and reserve lightpaths that are shared at different times with other applications. Several examples of these applications within areas, such as data mining and visualisation can be found within the realm of the OptIPuter project. We will also consider network situations were multiple network providers must work together in order to create end-to-end lightpaths. We will assume that providers will allow applications or their middleware to make lightpath reservations. As lightpaths typically do not use network layer data forwarding techniques, and rely on layer-2 or below technologies, access control to a lightpath becomes more difficult. When a lightpath needs to be specifically assigned to an application, it becomes in particular difficult to guarantee exclusiveness. During the course of this section we

6 This section is based on the first part of publication "Multi-Domain Lightpath Authorization using Tokens", Leon Gommans, Li Xu, Fred Wan, Yuri Demchenko, Mihai Cristea, Robert Meijer, Cees de Laat, Future Generation Computing Systems, Vol 25, issue 2, 2008, pp 153-160.

will see that network domains and applications can work together in different ways to make sure applications, which reserve a lightpath, actually get unique access to their reserved lightpath.
Hybrid networking concepts within networks, such as SURFnet6 [SURF], Internet2 Dynamic Circuit Network DCN [DCN], CA*Net UCLP [UCLP], G-lambda [GLAM] and GEANT2 Autobahn [AUTB] allow applications to reserve and use a lightpath on demand. Within these networks it is, however, unclear how particular applications can be given exclusive access to a reserved lightpath, whilst preventing other applications from using the same lightpath during its use. In this section, we show a token based access control mechanism that can be used for this purpose. Recent research and development projects, such as Phosphorus [PHOS] and Internet2 DCN aim at making network resources Grid middleware enabled. The token approach is being incorporated and tested in these projects.
A token provides a flexible mechanism that allows the right to access a lightpath to be associated with a request from an application. After a user (or application) requests access to a network resource, the network is able to recognize a token that enforces access across multiple domains. We will show how tokens can prevent other users or applications from gaining access to the same resource at the same time. The focus will be on the access enforcement ability of the network, its granularity, and ways how the network can create the associated context needed to enforce a token. We will only mention some of the policy-based decision types that domains typically make before they decide to grant access.
In the paper, we will first elaborate on the concepts around tokens. We will then briefly describe how these concepts were applied in various provisioning and access control enforcement models. We will end by briefly describing demonstrations during subsequent iGrid 2005 and SuperComputing (SC) events in 2005, 2006 and 2007.

3.4.2 The token as a concept in networking.

Referred by section 2.2.6, this section will elaborate on the concepts around tokens in the context of networking. Questions like "Why use tokens?", "What is a token?", and "How are tokens created and handled?" will be discussed.

3.4.2.1 Why use tokens?

Current optical network control and management plane implementations do not employ mechanisms that consider and enforce data-flows from individual application sessions. These implementations enable users to reserve and allocate a lightpath. After allocation, the application signals the network using protocols such as RSVP-TE [R5151] or XML/SOAP that it likes to use the lightpath. The allocation typically specifies a lightpath between two endpoint addresses, for example physical port numbers or IP addresses. The network typically assumes that the application component is directly connected to the specified ports. Most mechanisms will first authenticate and subsequently authorise the application user before allowing the user to make a reservation for time and bandwidth between the endpoints. Once completed, the network does

however not enforce the relationship between the user dataflow and the lightpath. The network assumes that the application will use the same ports as requested. It also assumes that no other applications will share the connection at the same time. These assumptions make authorised public usage of hybrid networks, offering lightpath services, more complicated. In addition, authorised usage becomes more complex when the reservation process involves multiple domains. In such cases, the downstream domain must trust the upstream domain that it forwards the intended flows. The pictured problem is not unlike making reservations on a multi-legged flight and selecting seats, without the presence of airport authorities and/or airline employees to enforce access to the intended plane and its seat. Without such enforcement, anybody could board the plane and occupy the reserved seat without the rightful person being able to prove his/her right to be seated on this flight. Airlines use boarding passes. In networks we propose to use tokens for the same purpose.

3.4.2.2 What is a token?

The word "token" is an overloaded term. The term is likely to create confusion if we do not define it in the context of our research. While the generic meaning of the word "token" is "a visible or tangible representation of something abstract", "a characteristic or distinctive sign or mark", the "security token" as it is defined in the Web Services Trust [WSTL] context actually means a security protected credential. Within our context, we therefore use the following general working definition for a token: "*The permission is a small piece of information that unambiguously references information providing the context of a specific lightpath session.*" Tokens are used as part of a security scheme, where its possession proves a right, when challenged during resource access control phase. Tokens are different from certificates and tickets, in the sense that a certificate carries multiple attributes in a specified format, and each attribute has a defined and explicit meaning. A ticket also carries attributes but its scope and validity is limited, and its format is application dependant. Tokens, certificates and tickets have in common that they are integrity protected and its authenticity is ensured by the issuer or signer. In comparison to a certificate or ticket, the meaning of a token is strictly abstract.

A token is obtained, carried and presented by a holder. The recipient must understand its abstract meaning. This understanding may be contained in the logic of recipients program and may be augmented by the authority before a holder presents a token. The same token may express different meanings when the holder presents it to multiple recipients. Authorities must therefore make all possible recipients aware of the relevant meaning of the token. This may be perceived as a disadvantage, however tickets or certificates recognition by multiple recipients require that their attributes must share an agreed meaning. The abstract nature of a token allows flexible usage in multi-domain lightpath provisioning scenarios. A token references a shared, context dependent meaning.

3.4.2.3 Authenticity of a token

A token must carry a proof of its authenticity. This can be achieved by using a secure message authentication algorithm (e.g., HMAC-SHA1) to calculate (part of) the content of the token that must be recognized by the recipient. The key, used in the algorithm, must either be shared between authority and recipient, or the recipient must have an exact copy of the token. In this way a trust relationship will be established between authority and recipient. If the digest used to generate and verify the token includes (part of) the service related context, the user will not be able to modify this context without invalidating the token. We will see that the token can be used at different layers. At the IP layer, the token digest can for example include the IP addresses, TOS value, etc. Modifying the destination IP address of the packet will invalidate the token. We will see that higher layers typically use a unique session ID as digest.

3.4.2.4 Tokens as part of an authorization sequence

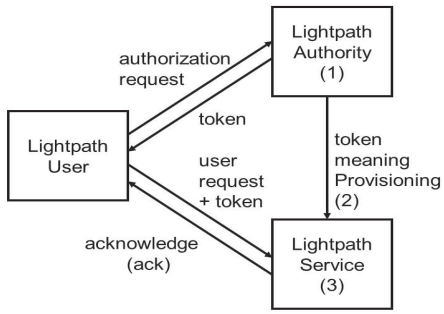


Fig. 3.4.1. The basic token sequence as an extension of the basic RFC2904 push sequence showing the position of the three provisioning process stages.

The presented solution is based on further development of the AAA Authorization Framework RFC2904 [R2904]. The push model, described in this framework, has been used in scenarios that implements network resource provisioning involving multiple domains. The provisioning process can be split into three stages [DEM8]: (1) reservation/authorisation, (2) deployment or activation, and (3) access or use/consumption. The reservation stage, which involves the user, may require (sometimes complex and time consuming) interactions to find, select, schedule and authorise the appropriate resources. In Section 4.5.1 we will explain that our implementation allows authorisation languages such as XACML [XAML] and SAML [SAML] to be used during these interactions. We will assume that resources can be committed after relevant authorisation decisions have been made. Subsequently we assume that the reserved resources can be associated with a common access control token at the end of stage 1. During stage 3, the token will be presented as a part of the network access request in each domain. At this stage, a token will be evaluated against the reservation context (meaning) stored during phase 2 inside a domain that is referred to by the token. Fig. 3.4.1 illustrates the extension to the RFC2904 push sequence for a token. The addition to this sequence is the part where the token meaning is provisioned by the authority. Note that, as its meaning is explicit, this part may not be necessary in case the authority replies a certificate or ticket.

Also note that Fig. 3.4.1 only shows the interactions needed to communicate authorisation, not the actual use of the lightpath by the user.

The above sequence is aimed at allowing a Lightpath Authority to be flexible in assigning a specific context to a commonly agreed token. The Lightpath Authority is involved in the reservation/ authorisation decisions made during stage 1. The deployment stage (2) performs token meaning provisioning where the reserved resources are typically bound to some reservation ID carried by the token.

We will refer to this ID as the Global Reservation Identifier (GRI) that will be described in more detail later. The Lightpath Service performs stage 3. Stage 3 is like checking the passenger boarding the plane. The possession of a token enables the passenger (i.e. a user accessing a lightpath segment) to be checked whilst boarding the plane. When checking in on the next leg, the same token containing the reservation number (playing the role of GRI) can be used to refer to a different "seat number" (the context describing the next lightpath segment). This brings us to the subject of multi-domain scenarios.

3.4.3 Tokens in multi-domain scenarios

Here we consider the role of a token during the handling of a request by authorities in a multi-domain scenario leading to stage 2. We will then look at how tokens can be enforced inside the service entities at stage 3. To allow multi-domain lightpath provisioning, the domains must interact in a coordinated manner. Here we distinguish two typical approaches: the chain and tree approach. The chain approach is typical for multi-domain network provisioning scenarios used amongst Network Service Providers. An example of this approach can be observed within the Internet2 DCN network, where Inter Domain Controllers (IDCs) operate as domain Lightpath Authority. We will elaborate on this scenario in section 4.5. In section 3.4.3.3 we will discuss the tree approach, typical for Grid scenarios. We will first discuss the chain approach.

3.4.3.1 Context provisioning & token creation via the chain approach

When a user during stage 1 requests an authorisation from a Lightpath Authority to use a particular lightpath in a typically multi-domain optical network, each domain's authority will apply some policy when evaluating a request. Policies may imply rules and/or conditions regarding the identity of the requestor, its authorisations, the existence, route and (optimal) availability of the requested path, priority of the request, etc. Each domain may have its own policy what will imply a specific domain related context to a decision that the token will represent. Fig. 3.4.2 illustrates interactions between major entities participating in a multi-domain lightpath provisioning chain approach scenario. The process is initiated by a user request sent to the domain A's Lightpath Authority. At this stage, a GRI is created by domain A.

The GRI, must be a globally unique identifier. It can either be implemented as a, large, randomly generated number, that can be considered as sufficiently unique, or as a domain-unique number concatenated with a unique domain identifier.

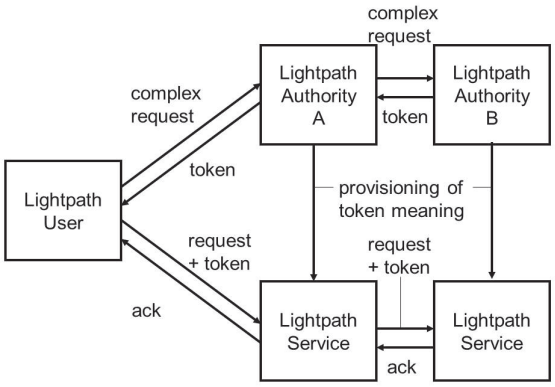


Fig 3.4.2. Provisioning a multi-domain chain of domains.

The GRI serves to identify a lightpath session across multiple domains. The GRI may also be used inside a domain to administer local resource details. The outcome of the policy decision process is either positive or negative. The negative result is logged and replied to the requester.

A positive result will cause a request to be administered and sent to the next domain (B) along the path. This request will include the GRI. A subsequent decision making process may again yield a positive or negative result. The GRI is used to administer the result and its details in domain B. A negative result is returned to the upstream domain (A). A positive result at this stage means that all previous domains can serve the request. Being the last domain in the chain, it also means that the entire request can be honoured. As a way of expressing this fact, a process in the last domain will create a token, by applying a secure message authentication algorithm (HMAC), to create a digital signature from the GRI using either a shared secret or trusted key of the last domain. To simplify secure context management, the token might just consist of the GRI and its signature. If not mentioned differently, we assume such a token in the remainder of the article. The signature might however be produced, by including part (or all) of the reservation context into its generation process for reasons discussed in 3.4.2.3. This step concludes stage 1 and will be followed by stage 2, where the reserved resource deployment/activation takes place. Stage 2 essentially means provisioning each domain Lightpath Service with the token or token-key and its associated meaning. This information allows token recognition and verification at the resource access stage (stage 3). At stage 2, the Lightpath Authority of domain B provisions the Lightpath Service of its domain. Lightpath Service B can use the GRI as an index to store the characteristics of the lightpath (bandwidth, time, ingress/egress points, etc.). Domain B must also, at end of stage 1, return the token or the key used to generate the token in the reply to domain A. Domain A will administer the reply, using the GRI as index. This will then also enable the service part of domain A to be provisioned.

The user will now receive from domain A the reply that includes the token (containing the GRI). At the agreed time, the user will signal the lightpath and include the token in the request. By comparing the token with the provisioned token (either provisioned directly or re-generating the token using the provisioned token-key), the Lightpath Service can quickly verify the validity of the token and provision the requested circuit. The GRI part of the token can be used to lookup the corresponding reservation context and token/token key that can be used for token validation. The request is then forwarded to the next domain where the same token is used as a means to perform access control to a set of different resources indexed by the GRI of the token. Note that communication during stage 1, 2 and 3 may be secured using a shared secret model or use a PKI based inter-domain trust infrastructure. This kind of security is considered independent of the security used to make the GRI authentic, i.e. creating the token.

3.4.3.2 The token context

As discussed, each domain in Fig. 3.4.2 may associate a different meaning or context to a token: the token may refer in domain A to bandwidth for a specified amount of session time between a specific pair of ingress- and egress ports. The information about ingress and egress ports will be different for domain B. Moreover, domain A may use a different time slot granularity than domain B. If A uses 1 min timeslots and B uses 5 min timeslots, then allocating a 12 min lightpath means 12 min in domain A, but may be translated to 15 min inside domain B. Allowing authorities to each provision a different service context to a token is an essential characteristic.

3.4.3.3 Context provisioning & token creation using the tree approach

In Grid environments, the network resource may be provisioned in the same way as any other Grid resource. Grid applications typically use a centralised scheduler as common authority for this purpose.

Fig. 3.4.3 shows the tree approach. In Grid environments resource reservation and scheduling is a part of the middleware functionality. In collaboration with Santa Clara University, the University of Amsterdam investigates the use of an elastic scheduler [NAIK] to reserve network resources. Part of the Phosphorus [PHOS] project researches the functions of the ISS/VIOLA [GRUB] meta-scheduler for finding optimal choices when co-allocating. network- and computing involving multiple resources domains. Additionally combined grid-network resources reservation may allow creating optimal mapping between grid jobs and required distributed computational resources with network performance limitations. This topic is subject of research in the G-Lambda [TAKE] and Phosphorus [PHOS] projects.

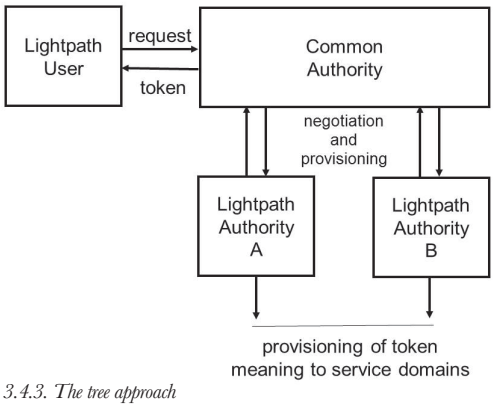


Fig 3.4.3. The tree approach

Within the tree approach, a common authority will negotiate with individual lightpath authorities along the path. If the common authority can resolve the request, it will provide a token to the user to indicate all involved domains are committed to provide the requested resource. Alternatively, each domain can create a token, where the common authority just passes it on to the user. In this case, the user needs to insert a number of tokens into the signal to use the lightpath, one for each domain. The feasibility of using the same provisioning and policy enforcement model for both approaches is part of our current research. We expect that tokens and the concept of a GRI can glue together both chain and tree style authorisation.

3.4.4 Access control granularity and enforcement layers

After the context is provisioned and stored inside the Lightpath Service, the service will wait for a service request to arrive for subsequent enforcement. When received, the GRI part of the token points to the context of the lightpath reservation stored by the Lightpath Authority. The service request with the token can be sent in a number of different ways:

1. *At IP packet layer*. Each IP packet is considered as an individual access request. At this level, the token is included inside each IP packet, e.g. inside the IP Options field of an IPv4 packet. This enables per packet enforcement. As per packet access enforcement is common in firewalls, we call this approach the firewall- or packet layer approach. At this level the token is typically a secure hash result of the context (e.g. content of IP packet header) and may even not contain a GRI.

2. *Control plane or Network layer path signalling* Each path signalling message, such as an RSVP-TE PATH message, contains a token. As RSVP-TE messages are sent at certain time intervals to keep the data-path alive, this kind of signalling will enable enforcement by keeping the path alive. An invalid token could cause a teardown of the path or could stop the forwarding of RSVP-TE messages by a Label Switch Router (LSR). Tokens could be placed inside a Policy_Data object as defined by RFC2750 [R2750]. We call this approach the path signalling approach.

3. *Service layer signalling* Service layer signalling typically employs an XML based protocol such as SOAP to implement a Web Service. A token can be part of the object exchange. The service application logic will determine if a single token exchange is sufficient to authorise the resource access, or that a token must be sent periodically to keep the circuit alive. We call this approach the service layer approach.

Note that each of these different approaches implies different levels of enforcement granularity. At IP packet layer, we have the finest granularity where each packet is subject to access control, whereas the approach at service layer could only be enforced once, i.e. when a lightpath is signalled when connecting.

Examples of these approaches were shown during subsequent Supercomputing events of 2005, 2006 and 2007 and during iGrid2005.

3.5 Summary

In this chapter we discussed the applicability of the Generic AAA architecture, interacting in ways described by a framework of authorization models named the Agent-, Push-, and Pull sequences. When placed in the context of networking, these models allowed us to reason about what is needed to implement authorization functionality. As such, our novel contribution of the Authorization Framework and Generic AAA Architecture models proved its value.

We first considered the applicability of the Agent model to authorize QoS bandwidth brokerage style scenario's as alternative for the Pull model based on the existing RSVP scenario. We motivated that researching applicability of the Token model was a promising direction to be researched. Compared to the other two models, the use of a token in the Push model would allow the separation of the request for network connection from the use of a network connection. We defined a token as a (small) list of attributes cryptographically bound to an AAA server acting as Attribute Authority without assuming a particular format (such as X.509). This has the advantage that a user, who possesses a token asserting access rights by using the token, can remain anonymous. We saw the token model evolve into essentially a combination of the Agent- and Push model. Considering multi-domain cases, we defined a token as *"A shared abstract permission that is presented as part of an access request in each domain"*. We showed that the token model can be applied in a number of scenarios involving different network technology layers and at different places in the network. A token essentially points to a meaning that has been defined in each domain the token is presented. As the token approach was not common in networking, our contribution helped to think about implementing such scenario's and reason why such scenario's would work better compared to other scenario's.

Along the way we showed how our Generic AAA Architecture components conceptually interact with other functions such as resource managers and network control functions and interfaces at different levels of the network. These interactions will be demonstrated in the experiments described in the next chapter.

Experiments with Authorization concepts

4

"Design is not just what it looks like and feels like. Design is how it works."

Steve Jobs (1955 - 2011)
Co-founder Apple

4 Experiments with Authorization concepts

In this chapter we will present experimental work performed based on the concepts and ways to deploy the concepts as presented in chapters 2 and 3. These concepts were implemented using the "Generic AAA toolkit" [AAATK]. This JAVA/J2EE based toolkit was developed over time at our research group. The toolkit supports distributed authorization decision taking using the concept of a Rule Based Engine and Application Specific Modules to translate policy based binary (yes or no) decisions into its meaning.

This chapter contains the experimental parts of publications that were introduced by sections of chapter 3. Table 4.1 shows what sections have been introduced by sections of chapter 3 (in parenthesis).

	Publication title	**Demonstrated at**	**Section (intro)**
1	Authorization of QoS path based on Generic AAA	iGrid 2002 (D)	4.1 (3.1)
2	Applications Drive Secure Lightpath Creation accross Heterogeneous Domains	SuperComputing 2004 (D)	4.2
3	Token Based path authorization at Interconnection Points between Hybrid Networks and a Lamda Grid	Gridnets 2005 (D)	4.3 (3.3)
4	Token Based Networking: Experiment NL101	iGrid 2005 (D)	4.4
5	Multi-domain lightpath authorization, using tokens	SuperComputing 2006 (D) SuperComputing 2007 (D)	4.5 (3.4)

Table 4.1: Overview of publications used for chapter 4 and corresponding sections.

As publications 2 and 4 were mostly about demonstrating an experiment, the (small) conceptual introduction part was for simplicity reasons not placed in chapter 3, but left with the publication placed in this chapter.

4.1 Agent sequence authorizing a single domain path[7]

As shown in table 4.1, this section is a continuation of section 3.1 that introduces the Generic AAA experiment. This section will describe a Generic AAA server processing a request using the Agent Sequence authorizing a QoS path. It uses a Driving Policy that calls Application Specific Modules with methods that manage a connection as a resource and provisions a Virtual LAN on a pair of switches.

4.1.1 Case Study

For the case study presented we will focus on scientific Grid users, who have a need for an on-demand high bandwidth QoS path. These users fall into a specific category of users that sets them far apart from regular Internet users. Classifying users for our purpose is a three-dimensional problem: The number of users can both be considered against the amount of bandwidth needed and the amount of destinations in need to be reached. In our case, we are targeting a possible solution for users that are both in need of very high amounts of bandwidth and are very limited towards the need to reach destinations. When considering the bandwidth usage only, these users may fall into the category C as mentioned in the article "The rationale of Optical Networking" [DLA3]. Also considering the fact that applications in this space may use application specific (adapted) protocols to reach optimal performance, this demonstration shows the authorization of a QoS path using layer-2 switches. With data transfer programs such as GridFTP in mind, the shown configuration allows the creation of a by-pass channel that can be used as the data-channel in parallel with the regular Internet connection, which in this case can be used as the control channel. The actual technology choice for this model was motivated by availability of layer-2 switch technology donated by Enterasys Networks. The concept used, however, may be applied to any layer-1, -2 or -3 technology that is capable of creating QoS path abstractions. The usage of the agent model effectively separates the control plane and data forwarding plane. As most of the intelligence exists in the control plane, deployment of less intelligent layer-1 or -2 equipment in the data forwarding plane seems a more suitable solution.
In our test-bed we have two 802.1Q VLAN switches interconnected by an optical fiber. Each switch connects two hosts, see Fig. 4.1.1. The network link is implemented as a Gigabit Ethernet connection. One of the hosts will send an AAA request for a BoD service via the regular Internet to the AAA server. The AAA server will fetch the BoD Driving Policy that describes the plan of actions in order to achieve the authorization and provisioning of the connection. After success a QoS path is provisioned as a private network between the requesting hosts and the targeted host.

---

7 This section is based on the experimental part of publication:
"Authorization of a QoS Path based on Generic AAA", Leon Gommans, Cees de Laat, Bas van Oudenaarde, Arie Taal, iGrid2002 special issue, Future Generation Computer Systems, volume 19 issue 6, pp. 1009-1016 (2003).

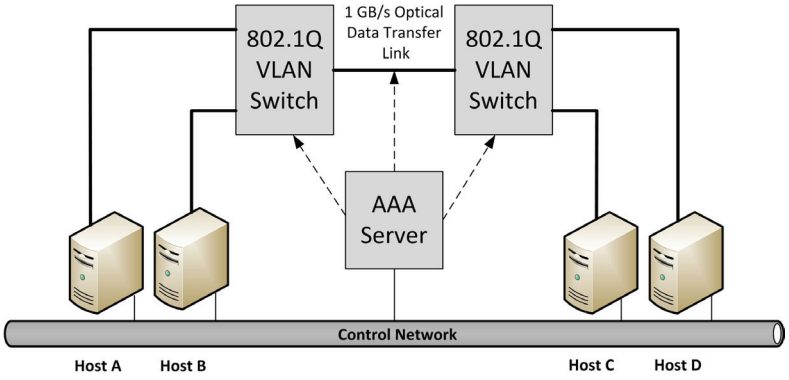


Fig 4.1.1 iGrid 2002 demo setup.

More implementation details are discussed in the following section.

4.1.2 AAA Messages

Both the top level data objects to be carried in an AAA protocol message and the various types of AAA protocol messages have not formally been defined yet. Data objects and message types must be defined in an abstract way without regard to encoding. The message types defined could be thought of as the Use Cases (in the Unified Modelling Language sense) for a generic AAA server. The data objects specify what kinds of information can be routed among the AAA servers in the Generic AAA infrastructure. We have chosen to express the different types of AAA messages in XML. The AAA protocol consists of request–reply pairs. XML schemas define these request–reply pairs. An XML schema provides a means for defining the structure, content and the semantics of an XML document. In order to keep the discussion simple, an AAA service request for bandwidth might look like:

```
<Request type = "BoD">
  <AuthorizationData>
    <Credential>
      <Type>simple</Type>
      <ID>JanJansen</ID>
      <Secret>#f034d</Secret>
    </Credential>
  </AuthorizationData>
  <BodData>
    <Source>100.10.20.30</Source>
    <Destination>110.1.2.3</Destination>
    <Bandwidth>1000</Bandwidth>
    <StartTime>now</StartTime>
    <Duration>20</Duration>
  </BodData>
</Request>
```

The AAA server returns a simple answer to the invoker whether the authorization has succeeded or not. This answer is added as a text node to the XML reply shown below:

```
<Reply type = "BoD">
  <Answer>
    <Message></Message>
  </Answer>
</Reply>
```

4.1.3 Driving Policies

In our policy language a Driving Policy is of the form 'IF (Condition) THEN (ActionList) ELSE (ActionList)'. The Driving Policies can be expressed as nested if–then–else structures, i.e. an if–then–else structure might be part of a Condition as well as part of an Action List.
As this is not the proper place to fully explain the policy language we will restrict ourselves to the example below in order to convey its expressiveness. A Driving Policy for an AAA server that accepts the above request for bandwidth might look like:

```
IF
(
  ASM::Authorizer.authorize(
  Request::AuthorizationData.Credential. Type,
  Request::AuthorizationData.Credential. ID,
  Request::AuthorizationData.Credential. Secret
                                    )
)
THEN
(
    IF
    (
     ASM::RM.CheckConnection(
       Request::BodData.Source,
       Request::BodData.Destination
                                    )
    )
    THEN (ActionList1)
    ELSE
     (Reply::Answer.Message = "Request failed :
       Bad source or destination"
     )
)
ELSE
(Reply::Answer.Message = "Request failed :
   Authorization not successful."
)
```

This Driving Policy calls for authorization by an ASM called 'Authorizer' that contains a public method 'authorize'. Three arguments have to be passed. The RBE can deduce that all arguments can be retrieved from the incoming request. If this call returns a false value the action

in the ELSE-part instructs the RBE to add a fail text node to the XML reply. After success the ActionList of the THEN-part is executed. This ActionList contains a single action, an if–then–else structure. A Resource Manager (RM) ASM is called to check if a route between both end points of the QoS path can be established. In this fashion, a number of pre-conditions are checked before the final call for bandwidth is made:

```
RV=
  ASM::RM.BoD
  (
    Request::SwitchData.Source,
    Request::SwitchData.Destination,
    Request::SwitchData.Bandwidth,
    Request::SwitchData.StartTime,
    Request::SwitchData.Duration
  )
```

By assigning the return value to the variable RV, the return value is available in subsequent actions of the Driving Policy.

4.1.4 Application Specific Module structure

In our BoD demonstration, we implemented an RM ASM needed to deliver the BoD service. An RM ASM is responsible for tracking the state and the discovery of the resources. The RM ASM also guarantees QoS by avoiding oversubscription of the GigE network link.
For 802.1Q, the RM ASM administers the used VLAN tags to form the QoS path between the hosts.

The RM ASM has also awareness of the destinations based on the primary IP address of the end station and it is able to discover the switch port the secondary interface is connected to. As such, the AAA server does control a simple QoS path within a single domain.

4.1.5 Conclusions and future work of experiment

The demonstrated case represents a first step into the usage of Generic AAA for authorization within a single domain. In this example, a QoS path is authorized using a set of Driving Policies executed by an RBE involving ASMs.
Generic AAA mechanisms should allow the support of many combinations of different types of applications. Generic AAA ultimately must support for example the combined authorization of a pay per view movie including QoS path based delivery across the network and a pizza to go along with it. If one of these items cannot be delivered, the entire transaction should not be authorized to proceed. Although such vision may not be realized anywhere soon, it does set the direction for future work.
The next steps will be to put research into creating multi-domain scenario's involving the

pictured individual and Partial Control models to form more comprehensive networks. The path discovery functions that indicate which AAA servers should be contacted does play an integral role. In this respect the emerging ASON technologies [ASTN], aimed at end-to-end provisioning of optical connections, are of interest. Further research will also be put into building complex decision networks where items such as scalability, stability and performance will be investigated.

4.2 Agent Sequence authorizing a multi-domain path[8]

This section shows an authorization scenario where AAA servers act in the Agent sequence model. AAA Agents perform network path authorization decisions based on interactions with the Service Plane, implemented by Nortel's Dynamic Resource Allocation Controller [DRAC]. DRAC is a network provisioning system that was originally developed by the Metro Ethernet division (MEN) of Nortel Networks. This division was research partner in the SURFnet GigaPort Research on Networking project. Its objective was to show how a DRAC based "Grid Networking Service" could provide a service based on an authorization transaction.

We realize an open, programmable paradigm for application-driven network control by way of a novel network plane — the "service plane" — layered above legacy networks. The service plane bridges domains, establishes trust, and exposes control to credited users/applications while preventing unauthorized access and resource theft. The authentication, authorization, and accounting subsystem and the dynamic resource allocation controller are the two defining building blocks of our service plane. In concert, they act upon an interconnection request or a restoration request according to application requirements, security credentials, and domain-resident policy. We have experimented with such service plane in an optical, large-scale testbed featuring two hubs (NetherLight in Amsterdam, StarLight in Chicago) and attached network clouds, each representing an independent domain. The dynamic interconnection of the heterogeneous domains occurred at Layer 1. The interconnections ultimately resulted in an optical end-to-end path (lightpath) for use by the requesting Grid application.

4.2.1 Introduction

Independently managed network domains have long been capable to inter-connect and implement inter-domain mutual agreements among service providers. Such agreements are typically reflected in policies negotiated via protocols and interfaces such as the Border Gateway Protocol (BGP) or the External Network-to-Network Interface (E-NNI). In Grid networks, users and applications need to gain greater control of network resources for them to exploit their atypical traffic patterns and meet their throughput/latency requirements. There is therefore a need to revisit the inter-connect process [JONE] between domains in the sense that:

---

8 This section is based on publication:
"Applications Drive Secure Lightpath Creation across Heterogeneous Domains", Leon Gommans, Bas van Oudenaarde, Freek Dijkstra, Cees de Laat, Tal Lavian, Inder Monga, Arie Taal, Franco Travostino, Alfred Wan, "", IEEE Communications Magazine, Feature topic Optical Control Planes for Grid Networks: Opportunities, Challenges and the Vision, vol. 44, no. 3, March 2006 © 2006 IEEE.

a) Adjacent domains are no longer guaranteed to render a transport service at the same OSI Layer. Furthermore, they do not necessarily reciprocate the same peering protocol (whether it is BGP, E-NNI, or proprietary)
b) At the time the network resources are requested, there may not be inter-domain mutual agreements in place other than best-effort transit
c) Some credited users (or software agents on their behalf) are afforded the choice to source-route their traffic across participating domains.

A domain is an independently managed network cloud exposing a set of ingress and egress points associated with Service Specifications. Provisioning an optical end-to-end path while crossing different domains is quite a challenge [OUD5]. As it can be expected in a multi-domain scenario, authentication, authorization, and accounting decisions may differ in method, protocol, and policy among the domains that the end-to-end path crosses. As well, the optical control planes may differ among the multiple domains. It becomes crucial to establish common syntax and semantics for accessing network resources.
In this section, we present a provisioning architecture that has the ability to integrate different approaches for authentication, authorization, accounting, as well as different styles of control over network resources. We reduce this architecture to practice via a "service plane", a new software stratum that resides on top of control planes such as ASTN, GMPLS, and JIT. The service plane encompasses software agents for the Authentication, Authorization, Accounting (AAA) and Grid Network Services software agents. Together, these agents allow users (and applications on their behalf) to negotiate on-demand network services such as low latency connection, high throughput transport, network knowledge services, and third party services. At the same time, the autonomous network domains can strictly enforce admission and usage policies and establish necessary trust amongst neighboring domains in order to avoid resource theft.
According to the telecommunications management network (TMN) model, the service plane belongs in the Service Management Layer (SML). Furthermore, the style of AAA chosen conforms to [R2903, R2904, R2905]. The paper continues with the description of the provisioning and trust architecture. Throughout section 4.2.3, we show how the service plane brings this model into practice, featuring the AAA agent and the Grid Network Services agent. Section 4.2.4 focuses on selected, distinguishing behaviors and dynamics for the service plane, such as restoration across domains. Section 4.2.5 focuses on the actual trans-continental optical testbed, which was used to prove and showcase the features of the service plane (as shown at the Supercomputing 2004 Conference in Pittsburgh, PA, USA), followed by conclusions and directions for further research.

4.2.2 A new provisioning model

Grid users are accustomed to allocate and relinquish some virtualized sets of computational, storage, and/or visualization resources. They do so with a high degree of automation, using software feedback loops and schedulers taking the place of GUI portals and operators.
In many Grid scenarios, the network element turns out to be a resource as important as computation and/or storage. As such, Grid users require the same level of control towards

subsets of well-defined amounts of network resources for the duration of a specific Grid task.
A chief goal of our service plane is to turn the network into a virtualized resource that can be acted upon and controlled by other layers of software, be it applications or Grid infrastructures (e.g., a community scheduler). In other words, the network becomes a Grid managed resource much as computation, storage, and visualization are.

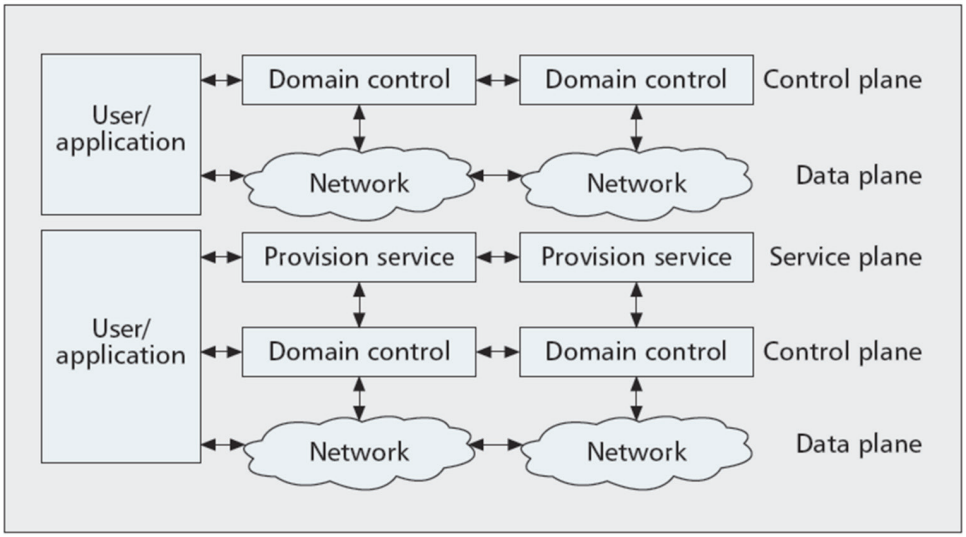


Figure 4.2.1. Top: the current model encompasses a control and network layer, allowing network providers to negotiate network capabilities. Below: The presented provisioning model adds a service plane, which allows additional application driven network control.

Layered upon the (optical) network control plane (see Fig. 4.2.1.), the service plane is typically concerned with path allocation, optimization, monitoring, and restoration across two or more domains. A service plane must be designed to be extensible from the ground up. It should allow adaptation of various control plane interfaces and abstract their network view or element set into the service plane. Examples of underlying control planes are: ASTN, GMPLS, JIT, etc. Each control domain has exclusive control of its resource and is typically able to signal neighbouring domains to create end-to-end paths on behalf of the involved Service Providers.
The AAA and Grid Network Services agents are the two key ingredients of the service plane. For the moment, we assume that each domain has implemented such agents.
A Grid Network Service agent advertises network capabilities to its (trusted) neighbours. In such way, each agent is able to generate a complete topology view in order to construct future optical end-to-end paths. Our agent can optionally support source based routing to select a preferred path through the individual optical clouds.
If a Grid Network Service agent wants the neighbouring instances to create a path, it requires a proper authorization token. The AAA part of the service plane obtains authorizations from multiple administrative domains. Before any path can be provisioned, all authorizations will need to be collected. We used our model in a GLIF-like context [DEFA]. Here parties are able join the federation if they allow some user controlled access to their network resources. The end-to-end

provisioning is split into a two-phase commitment process, whereby first the authorization is handled and resources (e.g. switches, and links) are reserved. Secondly, the actual commitment follows. Out of many approaches and sequences possible, we have chosen to adopt a RSVP-like signaling mechanism between Grid Network Service instances. The agent sequence is chosen with regard to AAA (per AAA Authorization Framework [R2904]).
Since the path negotiation transits across multiple domains, the inter-domain trust model is a fundamental aspect to the overall provisioning strategy. There is a peer-to-peer relationship among the AAA servers representing an organization or domain. Furthermore, there is a trust relationship between an AAA agent and the Grid Network Service agent in each domain. Once the User is authenticated and authorized by the AAA agent of the source domain, this AAA agent represents the User during the setup process. Path establishment has been accomplished by means of transitive trust. This model corresponds to the Chained Partial Control model as described in [GOM3]. The requests are authorized because the requestor is known and trusted and the resource policy conditions are met. In our model, we used a token mechanism to ensure the authenticity of a request. Managing trust by means of distributing corresponding key material can be done in different ways. As we focused on the authorization aspects rather then security aspects, we assumed the existence of some proven mechanism that will ensure safe delivery, storage and usage of keys.

4.2.3 Building the Service Plane

The following section will outline the Grid Network Services and AAA agents.

4.2.3.1 Grid Network Service agent

In our experiment, Nortel's DRAC implemented the Grid Network Service agent. Each participating network domain had one or more instances of the DRAC running. In case multiple instances of DRAC are running in a single domain, a master instance is elected. This DRAC master instance manages the domain and the inter-domain connectivity through peer messaging. DRACs core framework includes services like a policy engine, a topology discovery engine, workflow utilities, inter-domain routing facilities and smart bandwidth management fixtures.
DRAC exposes an API allowing coupling with applications. The interface to applications is bi-directional, enabling network performance and availability information to be abstracted upwards toward the application. Applications can request network services through this API. Applications can for example request a "cut-through" (high bandwidth, low latency) service allowing applications to bypass Layer 3 and directly transfer data over Layer 1 connections. Applications can further specify if they want this service on demand or via a time-of-day reservation. This kind of functionality is deemed especially valuable for the data-intensive applications used in research networks.

4.2.3.2 The Generic AAA service agent.

An implementation of a service using the University of Amsterdam's Generic AAA toolkit created an agent that was used to determine if a call from an authenticated neighbour is authorized to invoke DRAC. For a complete end-to-end path, the authorization is spanning multiple authorization decisions by autonomous AAA servers. The generic AAA based service agent invokes DRAC via an Application Specific Module (ASM). Such module performs application specific services such as interpreting the meaning of parameters passed to it (see [R2903]). The service itself may be provided by equipment external to the ASM, in our case DRAC. In such case, the ASM communicates with the service via a well-known protocol (in our case TL-1).
DRAC is consulted to determine the network semantics. In the source based routing approach, a sub-solution of the end-to-end path is passed from neighbour to adjacent neighbour. This sub-solution contains DRAC pseudo objects, which reflects current path and suggested connection point solving the end-to-end path.
We implemented the Chained Partial Control model [GOM3] based on the concept of a driving policy running in each domain. A driving policy is a nested if-then-else structure executed by a Rule Based Engine (RBE, [R2903]), which is part of the Generic AAA toolkit. Its main task is to describe which pre-conditions have to be checked before actions, needed to fulfil an incoming AAA request, are delegated to ASMs. In general, these ASMs might apply their own policies and protocols. The type of incoming request determines the type of driving policy that the RBE should execute inside the AAA server. Every type of incoming request is assigned a driving policy. The RBE retrieves the driving policy from a Policy Repository managed by the administrator of a network service domain.

4.2.4 Service Plane Features in Focus

In the following section, we explore distinguishing features of our service plane. While the examples given in this section refer to the network layout shown in fig. 4.2.2, the principles are generally applicable.

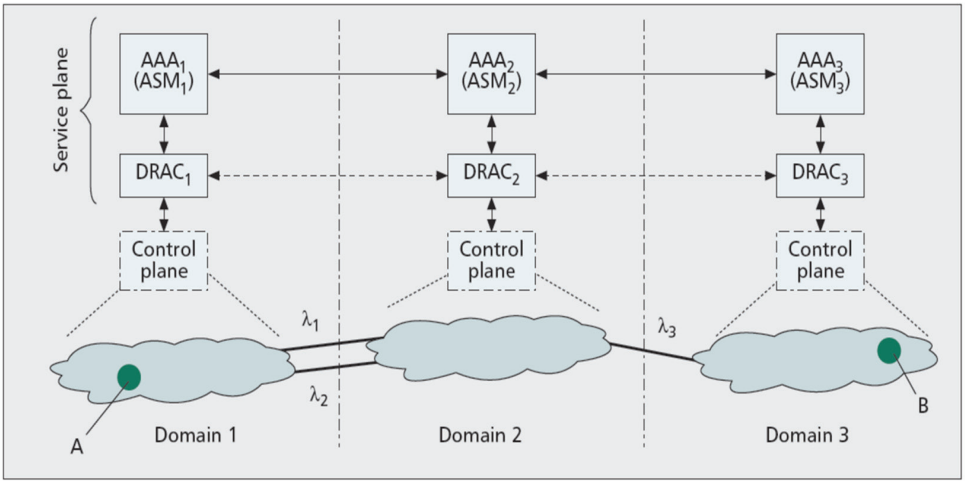


Figure 4.2.2 A network layout extending across three domains.

4.2.4.1 Message exchange in Multi-domain provisioning

Without loss of generality, we assume that the AAA_1 server in the domain of a source node A (see Fig. 4.2.2) receives the request by a User (or an application on its behalf) for a connection between network nodes A and B. The request is depicted in Table 4.2.1.

```
<AAARequest type="AAA_01" >
     <AAA>
       <Authentication>
         <Name>Joe User</Name>
         <Attribute AttributeId="token">
             SSBhbSBhbiBvcmRpbmFpcnkgdXNlcg==</
Attribute>
       </Authentication>
     </AAA>
     <DRAC>
       <Xfer>
         <SrcId>10.1.1.120</SrcId>
         <DestId>10.1.3.130</DestId>
         <Amount>1000</Amount>
         <TimeSpan>3600</TimeSpan>
       </Xfer>
     </DRAC>
   </AAARequest>
```

Table 4.2.1. Connection request from a user.

The content of the User's request is divided into two parts. The first part is enclosed between AAA-tags and contains the information concerning authentication, authorization and accounting. The second part enclosed by the DRAC-tags contains the information the DRAC needs. Typical information for $DRAC_1$ are the two network nodes `(SrcID and DestId)`,

the amount of information to transfer in Mbytes (`Amount`), and the time span within which the transfer must take place (`TimeSpan`).

The provisioning process encompasses a series of actions and messages exchanges between AAA servers along a path between source node A and destination node B. Server AAA_1 checks whether it can authenticate and/or authorize the User. When successful, AAA_1 extracts the information between the DRAC-tags in the request and passes it on to $DRAC_1$ by way of the Application Specific Module ASM_1. $DRAC_1$ is asked to define a connection between A and B. The reply from $DRAC_1$ to ASM_1 contains the specification for a request that must be forwarded to the next AAA server in the provisioning process. The message sent to AAA_2 by AAA_1 has a similar structure to the User's request as seen in Table 4.2.2.

```
<AAARequest type="AAA_02" version="0.1" >
     <AAA>
       <Authentication>
         <Name>AAA@science.uva.nl</Name>
         <Attribute AttributeId="token">
           SSBhbSBBQUEgc2VydmVyIDE=</Attribute>
         </Authentication>
       <SessionID>12335</SessionID>
     </AAA>
     <DRAC>
       <DRACXfer>
         <SrcId>10.1.2.121</SrcId>
         <DestId>10.1.3.130</DestId>
         <XferAmount>1000</XferAmount>
         <TimeVal>3600</TimeVal>
       </DRACXfer>
     </DRAC>
   </AAARequest>
```

Table 4.2.2. Message sent to AAA_2 by AAA_1.

The new attribute for the authentication and the new source node (`SrcId`) are worth noting. The token of the Attribute-tag identifies AAA_1. The new source node determined by $DRAC_1$ represents an ingress point into the next domain, i.e. the domain where λ_1 enters. $DRAC_1$ determines the first part of the path, an intra-domain connection between node A and an egress point, λ_1. This is a decision based on topology information.

The communication among DRAC agents keeps the topology information up to date. When AAA_1 needs to authorize access to λ_1, it is not evident who owns this connection. Here it is assumed that λ_1 is owned by AAA_2 or by a third party that AAA_2 knows about. AAA_1 must therefore first contact AAA_2 to obtain authorization.
This process continues until the last server in the provisioning process, AAA_3, with the help of $DRAC_3$, establishes a connection with destination node B. The reply from AAA_3 travels back along the path established. Upon receiving a positive answer, each AAA server commits the intra-domain connection determined earlier on and makes the necessary preparations for accounting. Finally, AAA_1 commits the connection between node A and λ_1.

4.2.4.2 Driving Policy

An AAA server's Rule Based Engine is guided by a driving policy that describes the actions to be taken by the AAA server upon receiving a request. This process is governed by a driving policy residing in the Policy Repository of AAA_1 (table 4.2.3).
The first action prescribed by the driving policy is the authentication of the User. Authentication is an example of a service provided by an ASM. Arguments for the function `ASM::Authenticate` are retrieved from the incoming request. The notation refers to the Authentication-tag in the request which has an attribute called `AttributeId` equals 'token'. This token is passed to the authentication function. If the authentication is successful, the requester is authorized, using function `ASM::Authorize` (see section Authorization for more details).
When all preconditions for the provisioning are fulfilled, a session needs to be created to keep track of the provisioning process. The function `AAA::GetSessionID` creates this session. Provisioning is initiated with the call, `ASM::DRAC.Setup`. In case of a multi-domain setup, the new source node returned by $DRAC_1$ is key information. Beside the data from the User's request, $DRAC_1$ applies the internal topology knowledge to choose lambda λ_1. It returns the endpoint of λ_1 in the next domain. This endpoint is used by the RBE to determine the next AAA server in the process, `AAA::GetDomainServer`. Each DRAC agent keeps track of the provisioning process by generating a handle and a session identifier, which are needed in future communication with the DRAC agent.
It is up to $DRAC_2$ to continue the provisioning process from the endpoint of λ_1. It decides to extend the path via λ_3 to the domain of the destination. This decision is also based on the possibility to arrange an intra-domain connection between λ_1 and λ_3. In the request sent to AAA_3, the new `SrcId` equals the endpoint of λ_3. The first phase of a successful provisioning process ends with a path between the end node of λ_3 and destination node B.
AAA_3 arranges accounting and replies to AAA_2. Upon receiving a positive answer, all AAA servers along the path commit the provisioning to their DRAC agent and make arrangements for accounting. Therefore, the policy specifies a call to `AAA::PrepareAccounting`. Making preparations for accounting involves creating a record that contains the session identifier together with the information returned from the call to the DRAC agent.
Other than a reply indicating success, the User is given a session identifier. With such an identifier, the User can intervene on the lifecycle of the session, e.g. by requesting an earlier termination.
.

```
if ( true )
  then (
    // store <Name> and <Attribute> values in local variables
    userName = Request::AAA.Athentication.Name;
    userAttr = Request::AAA.Athentication.Attribute[AttributeId#token];

    // store array returned in local variable K2
    K2 = ASM::Authenticate( userName, userAttr );

    // K2[0]: indicates success or failure
    // K2[1]: new key returned after successful authentication
    if ( (K2[0] == "SUCCESS") &&
         ASM::Authorize( userName, K2[1], "DRAC", "Setup" )
    )
    then ( // start the setup

      sessionId = AAA::GetSessionID( userName, userAttr, K2 );

      // store array returned in local variable O4
      O4 = ASM::DRAC.Setup( Request::DRAC.Xfer.SrcId,
                            Request::DRAC.Xfer.DestId,
                            sessionId,
                            Request::DRAC.Xfer.Xfer.Amount,
                            Request::DRAC.Xfer.TimeSpan );
      // O4[0]: indicates single or multi-domain solution
      // O2[1], O2[2] internal variables applied by DRAC
      // O4[3]: end point of a lambda determined by DRAC
      if ( (O4[0] == "SUCCESS") ) // single domain solution
      then ( AAA::PrepareAccounting( sessionId, O4 );
             Reply::Answer.Message = "permit";
             Reply::Answer.SessionId = sessionId
      )
      else (
        if ( (O4[0] == "PENDING") ) // multi-domain solution
        then (
          // O4[3]: end point of a lambda determined by DRAC
          destAAA = AAA::GetDomainServer( O2[3] );

          R =  AAA::SendAAARequest(
                               AAA::GetAuthNToken(),
                               O2,
                               destAAA );
          if ( (R == "PERMIT") )
          then (
            // O2[1], O2[2] internal variables applied by DRAC
            ASM::DRAC.Commit( O2[1], O2[2] );
            AAA::PrepareAccounting( sessionId, O2, R );
            Reply::Answer.Message = "permit";
            Reply::Answer.SessionId = sessionId;
          )
          else (
                ASM::DRAC.Rollback( sessionId );
                Reply::Answer.Message = "deny"
          )
        )
        else (
                ASM::DRAC.Rollback( sessionId );
                Reply::Answer.Message = "deny"
        )
      )
    )
    else ( Reply::Answer.Message = "deny" )
  )
  else ()
```

Table 4.2.3. A driving policy.

4.2.4.3 Authentication and Authorization Extension Mechanism.

By means of Web Services, a User can find out what kind of authentication data has to be carried in the message (message authentication). Once authenticated, the User needs to be authorized. This is the act of determining whether a particular right can be granted to a requesting entity with a particular credential. XACML (eXtensible Access Control Markup Language), an OASIS standard for expressing and evaluating access control policies, contains a usage model [ANDE] where a Policy Enforcement Point (PEP) is responsible for protecting access to one or more resources.. The PEP sends a request to a Policy Decision Point (PDP), which evaluates the request against policies and attributes and produces a response. The PDP in the XACML model applies its own policies. It is the AAA server's Rule Based Engine that resorts to an Application Specific Model that acts as a PEP in the XACML usage model. This is indicated by the `ASM::Authorize` call in the above listed driving policy.

4.2.4.4 Automatic Lightpath restoration.

If somewhere along the optical end-to-end path a link goes down, an alternative link might be arranged automatically in order to restore the lightpath. The control plane often recovers an intra-domain link failure. In all other cases, the link failure triggers an alarm condition inside the DRAC agent. The DRAC agent responsible for the broken link provides its AAA server with information about an alternative link.
The steps the AAA server subsequently takes are rather similar to those of an ordinary setup, except that no new session identifier needs to be generated. Accounting consequences of an alternative link during operation mode might be that the transport over the alternative link is more expensive. A reasonable strategy is to charge the transport over the alternative link according to the prior arrangement.
In case no restoration can be established within a small time span, the initiating AAA server should be informed that the session corresponding with the transfer has ended. A request to stop the transfer between source and destination node can travel backwards to the AAA representing the User. This is facilitated because in a request exchanged among AAA servers the requesting server identifies itself and provides the server downstream with a session identifier.

4.2.5 Experimental setup

The provisioning model was demonstrated during the SC2004 conference, held in November 2004 in Pittsburgh, PA, USA. Three optical domains, NetherLight, StarLight, and OMNINet were part of the experimental testbed, representing Domain 1, 2 and 3 respectively in Figure 4.2.2. NetherLight is the optical infrastructure in Amsterdam. The University of Illinois at Chicago manages StarLight, and Nortel, SBC Communications Inc. and Ameritech support OMNINet. In Fig. 4.2.2 the testbed is depicted showing the three optical networks, each considered as a single administrative domain.

One should interpret the inter-domain lambdas as a collection of optical entities necessary to establish a connection between peer networks. For instance, the connection between NetherLight and StarLight is a collection of SONET based optical Exchange Points [DIJK] and transit provider links for long distances.
In each domain, a single DRAC agent was given responsibility for the setup of intra-domain connections. It also connects ingress points to egress points in the domain under its control. In this testbed, we simulated a link failure by switching off a port on one of switches providing inter-domain connectivity, thus generating an inter-domain failure event. We then measured end-to-end link restoration times across the three domains shown in Fig. 4.2.2. The elapsed time for both detection and restoration of the inter-domain failure (Fig. 4.2.3) was in the order of a minute. Although there are several ways to optimize such times, this is already a vast improvement of the usual hours to days required to restore a connection using conventional means such as phone or email.

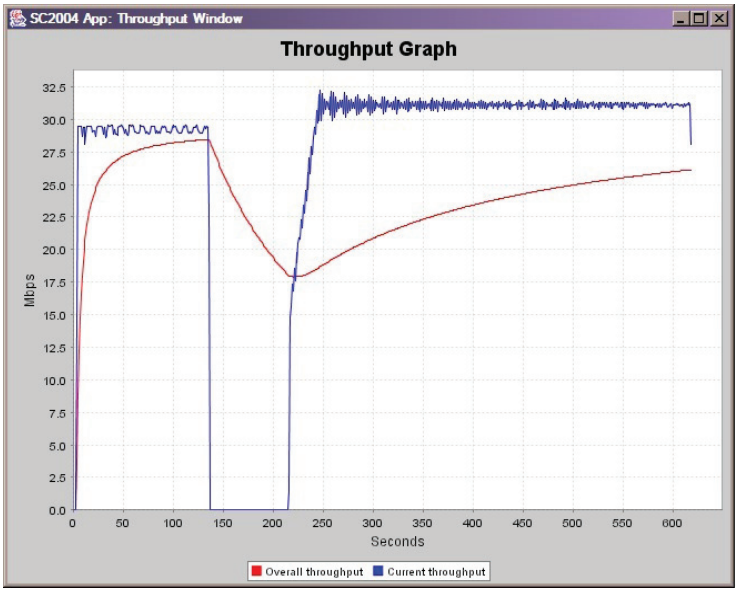


Fig 4.2.3. The current throughput line (top) shows the tie interval at [140,210] s that is required for the service plane to detect and recover the simulated interdomain failure. The bottom line represents the overall throughput.

4.2.6 Conclusions from experiment

If somewhere along the optical end-to-end path a link goes down, an alternative link might be arranged automatically in order to restore the lightpath. The control plane often recovers an intra-domain link failure. In all other cases, the link failure triggers an alarm condition inside the DRAC agent. The DRAC agent responsible for the broken link provides its AAA server with information about an alternative link.

The steps the AAA server subsequently takes are rather similar to those of an ordinary setup, except that no new session identifier needs to be generated. Accounting consequences of an alternative link during operation mode might be that the transport over the alternative link is more expensive. A reasonable strategy is to charge the transport over the alternative link according to the prior arrangement.
In case no restoration can be established within a small time span, the initiating AAA server should be informed that the session corresponding with the transfer has ended. A request to stop the transfer between source and destination node can travel backwards to the AAA representing the User. This is facilitated because in a request exchanged among AAA servers the requesting server identifies itself and provides the server downstream with a session identifier.

4.3 Token Sequence authorizing network level access[9]

As shown in table 1 of the introduction of chapter 4, this section describes the experimental part of the concepts contained in section 3.3. This section describes a experimental setup developed in our group by Mihai Cristea. The setup uses an Intel IDXP 2850 Network Processor development platform as a network switch. The switch was micro-programmed to recognize tokens inside IP packets. Based on recognition and validity of a token, the switch would forward a packet to a particular port or forward it to a default port, or drop the packet.

Section 3.3.5 ends with the recognition that the AAA sequences suffer from performance issues when a request needs to be authorized and its decision implemented at the same time (as is the case with the agent- and pull model). This section motivates that decoupling the taking of a decision from the usage of the decision is a way to solve timing issues when collecting authorizations from different places. The experiment shows that if a pre-arranged token represents the authorization, it can be enforced in real time at packet level at high performance, as such proving its viability.

4.3.1 Token model at Interconnection Points

Our experiments, referenced in section 3.3.5, assumed direct peering between networks when they were chained to create an end-to-end path. All control and management functions are implemented inside each network domain. Networks signal path authorization messages using a network of AAA servers. We will now consider situations where a path will explicitly traverse a Lambda Grid and their ICPs. As noted in section 3.3.4, ICPs need a switch to select a path. A switch needs control input in order to choose the path. When applying the token model to control an ICP switch, the token can be handed in two different ways:

---

9 This section has been based on the experimental part of publication:
"Token Based path authorization at Interconnection Points between Hybrid Networks and a Lambda Grid", Leon Gommans, Cees de Laat, Robert Meijer, IEEE GRIDNETS2005 proceedings, ISBN 0-7803-9277-9. © 2005 IEEE .

Out-of-band – here the switch control function (or higher level function) accepts a token via a separate interface that selects a path within the switch. The control function checks the authenticity and integrity of the token. The embedded information inside the token is used to provision the switch.

In-Band – Tokens are inserted into the IP-packets that flow through a switch. The switch will base its forwarding decision on the integrity and authenticity of a token. A switch must be pre-provisioned with the right key-material and forwarding information. The key material is needed to allow checking of the token. The forwarding information is needed to determine the output port. A default port can be configured to forward IP packet-flows with no or invalid tokens. A default port could be connected to a best-effort network.

The token must be requested, authorized and created first. A client can make the request using web services mechanisms to an AAA server. This AAA server can act as a broker between a set of other AAA servers. It can collect the proper authorizations from the involved stakeholders in an arbitrary complex way. Recent research of several groups within the GLIF community suggests that workflow mechanisms could be helpful in automating such scenarios. We will not go into further detail here. We assume that an authorization for a path is obtained somehow.
Considering recent advances in high-performance network chip technology used inside switches, the in-band signalling approach deserves a closer look. As said, the in-band approach requires a switch to recognize tokens present in the data-flow. Tokens must be inserted into each datagram at some point in the network, e.g. at the ingress point of the hybrid ISP. The hybrid network will forward the datagrams to the switch of the ICP. Recognizing tokens requires a switch to perform cryptographic functions such as a Message Authentication Code calculation. These types of calculations can be performed at very high speeds, using the cryptographic functions present in the latest generation of Network Processor Units (NPUs). This feature makes application of a token-based switch feasible for application within an ICP. We use an Intel IXDP 2850 development platform to implement such a type of switch. The token-based switch is both programmed to recognize tokens and to generate and insert a token into a datagram.
There are several options to insert a token into a datagram. A straightforward way is to insert a token into the IP-options field. IP-options is defined in RFC791 [R791]. This field can be of variable length and resides in the IP header. An IP packet containing options should be forwarded unmodified by a router. A router does not need to understand IP options. This makes the use of IP options transparent to a connectionless routed network in case a token is for example invalid. When a token inside the IP options field is used to make a path decision and when the other IP header information is used to route the packet, the principle elegantly marries connectionless and connection-oriented networking.

Fig 4.3.1. Shows in more detail the network and control plane components around ICP A as shown in fig 3.3.4 of section 3.3.4. It is assumed that a user application will use Hybrid ISP A to connect to some service via a peering ISP. The shown control plane elements all use Generic AAA toolkit components to allow distributed policy driven decision taking. A Link Request Service (LRS) generates a Link Access Request (LAR) message to use a particular link. The ISP may for example provide this service.

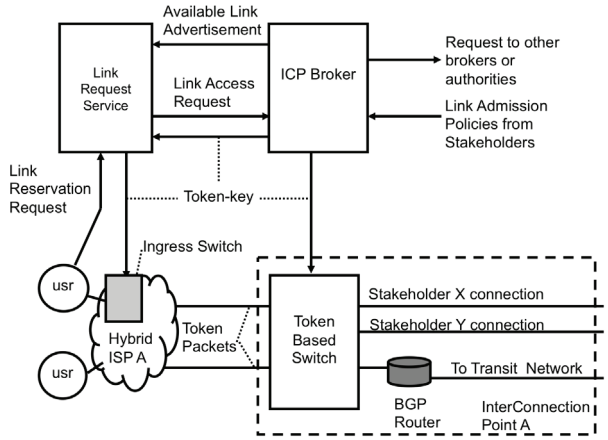


Fig 4.3.1. Token based framework for an Interconnection Point. This picture shows a token-based switch that is controlled by an AAA server that both accept link access requests from a user (via a web server) and link access policies from the link stakeholders

Amongst many possible policy driven scenarios, the LRS could first authenticate the user application and then offer a selection of appropriate links to this application. This information is based on information pushed by the ICP broker to the LRS. Link owners provision the ICP broker with link admission policies that will indicate who may use which links at what times. After the application decided which link it wants to use when, the LRS will compose a LAR defining a period of time and a specific link. This LAR will request one or more tokens from the broker. The tokens may be bound to specific timeslots. Before authorizing the request for the tokens, the driving policy of a broker may define that it needs to contact one or more other brokers along the end-to-end path. The LAR could contain several additional attributes including for example the Source and Destination IP (SIP/DIP) address of the stations or networks that want to communicate across the Lambda Grid link. The token switch can then recognize specific traffic, which it must treat in a connection-oriented fashion. With SIP and DIP, the user essentially asks a connection to be authorized between two classless IP addresses. After the ICP broker receives this information, the next section will explain how a token is created and inserted into the packets send to the token-based switch.

4.3.2 Token Creation

A token is essentially the result of applying a MAC algorithm on a number of fields of the IP-datagram as shown in fig. 4.3.2.

A MAC algorithm is a key based way to create a one-way hash. The key is called the token-key. As the result of a masked IP header field may yield the same information for consecutive IP

datagrams, one can avoid the generation of duplicate tokens by including (parts of-) the data-field into the MAC calculation.

Further research must motivate specific choices. A MAC algorithm provides data integrity and data origin authentication with respect to the originator of the message. An algorithm that is considered safe in creating a MAC for variable length messages is described in RFC3566 [R3566].

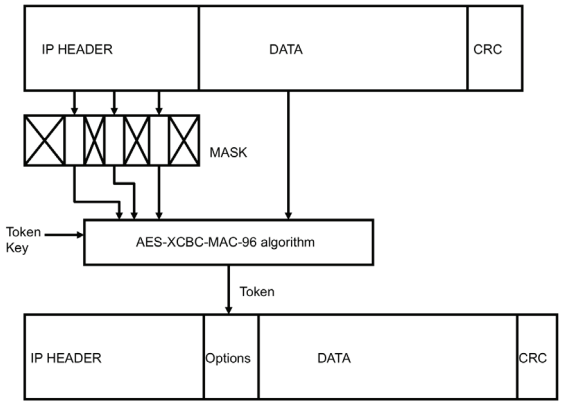


Fig 4.3.2. The token creation process masks parts of the IP Header and IP data field to create a unique token for each IP datagram using a key dependant MAC

If a MAC is created this way on an IP-datagram, the receiver of the MAC can be sure that a device holding the proper token-key generated the packet. We now call the IP-datagram containing a token, a token-packet. The token-key must only be provided to devices authorized to generate or check a token-packet. The token-key must only be valid for the time a link may be used. This will allow a link resource manager to provide precise control of the link usage. The ICP brokers, as shown in fig. 4.3.1, must distribute token-keys both to the token-based switches along the path as well as to the devices that insert a token into the token-packets. Fig. 4.3.1 shows that the ISP will insert the token in a token-packet on behalf of the user. Insertion could be performed, for example, on the ingress switch of the ISP. This approach is called in IETF terminology the "bump-in-the-wire" approach. An alternative would be the "bump-in-the-stack" approach, where the IP stack of the end-station would insert a token. When packets arrive at the token-based ingress switch, the token-key is used to generate a token according to fig. 4.3.2 and subsequently inserted into the IP options field.
The token-packet is then forwarded to an output port on the switch that (virtually) connects to a path provisioned by some control mechanism inside the ISP. This path leads to the ICP. The token switch inside the ICP will verify the token by applying a similar MAC algorithm with the same token-key on the token-packet. It will apply the same mask to the IP-header to extract the desired field and will combine this with (parts of-) the IP-data field. A MAC will generate a cipher that will be compared with the token inside the IP options field. If a match is found, the token

is authentic and the packet can be forwarded to a designated port owned by the stakeholder that was involved in the process that issued the token-key. This process effectively selects the authorized path in real-time. If the token does not match or if no token is present, the token-packet will be forwarded to a default port. This port could be connected to a border router that connects to a transit provider who, by definition, can forward the token-packet to its destination. A token-switch, using the Intel IXP 2850 employing 16 micro-engines and 2 crypto-units, is expected to handle token generation and recognition at speeds of up to 10 Gb/s.

4.3.3 Conclusions and future research

We described in section 3.3 the evolution of the Internet into a three-tier structure, where by-passing a transit network is common practice based on economic factors. Regulatory changes allowed any organization to own long distance transport connections, which created, through scientific collaboration, a Lambda Grid. From a scientific viewpoint, we assumed a desire by Lambda Grid link owners to authorize access to its resources for individual user communities. Within this context, we considered that:

- A token-based switch can select an authorized path at an interconnection point between a hybrid network and a Lambda Grid link in real time.
- Web service based components can be used to authorize link access using policy driven scenarios. The obtained authorization can then be transformed into a token-key, which can be used to generate token-packets whenever an authorized path must be accessed.
- A connectionless network is able to transparently forward a token-packet.
- A token-based switch can both connect to a connectionless and connection-oriented network. An IP-datagram without a valid token should be forwarded to the connectionless IP transit network. An IP-datagram with a valid token is switched to a particular connection-oriented link of a stakeholder. As such, we showed that a token switch effectively marries both networking paradigms.
- Current NPU technologies should be able to provide the necessary processing power to handle speeds up to 10 Gb/s.

We need to proof the presented concepts by conducting experiments with the IXDP 2850 NPU development platform. Many questions are still open for research: How could the initial address resolution process be performed best? How should IP fragmentation be handled? What is the optimal granularity for token-validity? Can tokens be re-used, grouped, shared etc. We saw that tokens could be bumped-into-the-wire at the ingress switch of an ISP. This will allow future research into firewall type functions for high volume streams. Can a modern network card, allowing TCP off-loading, be used to bump the token in the stack?

4.4 Token Sequence authorizing lightpath access[10]

This section describes an experiment performed at iGrid 2005, involving the Token sequence (combining the Push- and Agent Sequence) using the Generic AAA toolkit to handle the authorization sequence. Components of the toolkit interfaces with DRAC (Dynamic Resource Allocation Controller) to create a lightpath and performs decision taking to provide an application access to an optical connection by means of an optical switch. It shows the toolkit's ability the to communicate such decision via a simple token associated with an application oriented IP stream. This experiment was performed in collaboration with Nortel Networks that developed DRAC (see previous section).

This section outlines an experiment where tokens, associated with application oriented IP streams, authorize access to an optical lightpath during iGrid 2005. The experiment showed the practical viability of this mechanism to perform access control and resource management within optical networks. The experiment was conducted in collaboration with the Virtual Machine Turntable (NL-103) experiment, which used the mechanism to obtain access to continental and transatlantic optical network segments that connected Virtual Machine sites.

4.4.1 Introduction

Traditionally a client asks the access network owner to setup a network connection on its behalf. In the telephone infrastructure and the current Internet this is possible because a common understanding of the requirements exists: e.g. a 64 kbits/s constant bit rate leased channel or a best effort service. In cases where application specific requirements occur, path setup is much more complicated. There are trivially two approaches to setup an application specific network service: ask the network owner to do it on your behalf, or do it yourself. In the former case the network owner must be able to understand quite exotic requests possibly with an ad-hoc demand pattern. In the latter case application programmers must be able to interact with networking equipment and the applications must be authorized to do so. This is a realistic option as work performed in for example the GGF GHPN-RG [GHPN] found that web and grid services technologies in combination with proper abstractions of optical switches provides practical technologies and understandable concepts for the application programmers. We have continued this approach and in the subsections of section 4.4 we describe results of an experiment of a prototype of a path setup mechanism that supports application specific networking with a focus on the authorization aspects.
GFD.38 [GFD38] and RFC2904 [R2904] describe three fundamentally different authorization sequences between a User, a (network) Resource and an Authority. The sequence, where a user contacts an authority to obtain an authorization token, which is subsequently used to gain access to a resource, is called the push sequence. Unlike the other two sequences described in the

10 This section is based on publication:
"Token Based Networking: Experiment NL101", L. Gommans, B. van Oudenaarde, A. Wan, C.T.A.M. de Laat, R. Meijer, F. Travostino and I. Monga, iGrid2005 special issue, Future Generation Computer Systems, volume 22 issue 8, pp. 1025-1031 (2006).

Authorization Framework document (RFC2904), the push sequence is not commonly applied in low-level (optical) networking because, compared to the other two sequences, the push sequence requires the support of cryptographic algorithms and associated key management mechanisms. The other two, more commonly found sequences are the pull- and agent sequence. Using the pull sequence, the Policy Enforcement Point (PEP) embedded in the resource will outsource a decision to admit a user to a Policy Decision Point (PDP). After authorizing the request within the agent model, the PDP will provision the PEP within the resource with configuration information. Examples of the application of the pull- and agent sequences within networking can be found with COPS [R2748] and COPS-PR [R3084]. Considering the push model, tokens are mentioned in RFC 2753 [R2753] as an option of the RSVP protocol. RSVP is used for example to signal a Labeled Switched Path (LSP) in MPLS [R3031]. Tokens are also defined as extension of the SIP protocol [R3313] for IP-telephony. However, these protocols implement only the second half of the push sequence, i.e. the part where the authorization is used. How tokens are created, issued, assigned, distributed and used, is left to the implementation. These observations spurred our research into the characteristics and applicability of the push sequence in situations where time limited access is granted to a lightpath by means of a token. Within the context presented here, we defined a token as a cryptographically signed list of attributes. Signing ensures source authenticity and integrity of attributes contained within the token. We assume that key material, which creates trust between the authority and the resource, is distributed by some secure method based on some established relationship.
This short communication will first describe the rationale for investigating the push sequence in more detail. We will then describe the iGrid 2005 experiment setup and close with describing our experience and timing results.

4.4.2 Rationale

Tokens allow the decision process, which may involve complex and time consuming evaluation of attributes from different parties, to be separated from the usage of a token. Tokens can be handled in various ways after they have been created and issued to a user (or application). Such handling may support a number of different business-models. After being issued, a token may be exchanged or used in a mercantile system before being used to access a service. Tokens are also an effective way to manage resources. A token may represent an exclusive right to access a service at a particular time for a measured amount of time. These are all important reasons for us to research the application of the push sequence next to pull- and agent sequence. In the case described in this document, tokens are used as means to manage unique access to a network resource. Another reason for investigating the push sequence is that one may find other sequences slow. For example, at SuperComputing 2004 we showed [GOM61] that it takes approximately 75 seconds to authorize a new connection across a chain of three Generic AAA (Authentication, Authorization and Accounting) toolkit driven network domains, which used combinations of the agent- and pull sequence. We implemented an automated policy based mechanism that negotiated a back-up connection after the detection of a fibre-cut. The optical connections were managed by a state of the art high-level network management system from Nortel called DRAC (Dynamic

Resource Allocation Controller). Combined, a per-domain set AAA and DRAC components created the service management plane on top of the individual network control planes. Although 75 seconds is a vast improvement over traditional link restoration procedures using phone and/or email, this time may be large enough to cause timeout conditions in applications. One may for example expect a token to immediately authorize access to a pre-established backup connection.

4.4.3 Experiment description

In packet-based networks, tokens can either be sent in the wire (in-band) or next to the wire (out-of-band). In-band means that packets contain tokens to be recognized by the equipment handling the packets. Out-of-band means that an interface, which is separate from the packet-forwarding interface, accepts tokens via some signalling protocol. Within the realm of lightpath provisioning, we are experimenting with both methods [GOM3, OUD5, GOM5]. This experiment describes an out-of-band method. Our Token Based Networking demo used the Virtual Machine Turntable experiment (iGrid 2005 experiment NL-103) as an application to demonstrate the principles. The Virtual Machine Turntable demonstrated the move of Virtual Machines (VMs) between sites that are interconnected using dynamically configured lightpaths. The VM experiment uses the Linux based XEN monitor [XEN] from the University of Cambridge as the underlying OS. Within the NL-103 experiment, a Virtual Machine Traffic Controller (VMTC) governs the movement of VMs within a setup of VM sites. In order to pre-allocate a dynamic lightpath for a particular time-slot, the VMTC obtains one or more tokens from a Token Request Authority that governs usage of a lightpath between sites. In our case, a link between Amsterdam and

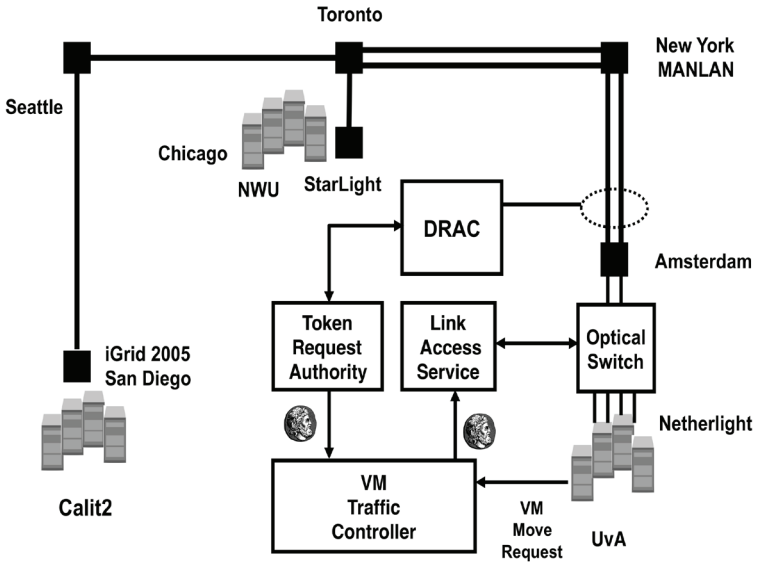


Fig 4.4.1. iGrid 2005 Setup.

Chicago was subject to this type of control. A token represents the right to use a particular dynamic lightpath at a specific time for a specific period. A single period is considered large enough to move a single VM. Fig 4.4.1, illustrates the setup used at iGrid 2005. Components of the Generic AAA Toolkit [AAAPR], developed at the University of Amsterdam, were used to implement the two components shown in fig 4.4.1 that exchange tokens with the VMTC. The function of the Token Request Authority (TRA) is to ensure that unique tokens represent a unique time-slot on a specific dynamically configured lightpath. Acting as PDP, the TRA uses a policy to coordinate requests with DRAC.

DRAC creates a lightpath connection between two end-points indicated by a command containing IP addresses and a time-interval. The connection is subsequently referred to by means of a handle ID. A token is bound to the attributes that describe the link.

DRAC is expected to create and remove a lightpath at the time the corresponding command specifies. Within iGrid 2005, the authorization and resource management function of the TRA was limited as it served a single VMTC application. The VMTC served a single setup of VM sites. Also, the VMTC controlled access to a single link between Amsterdam and Chicago routed via New York (MANLAN) and Toronto. DRAC controlled the connection between Amsterdam and New York; the other lightpath segments were configured. Therefore, only minimal functionality was implemented inside the TRA for the iGrid 2005 demo. The main function of the TRA was to signal DRAC to open the Amsterdam - New York connection and subsequently generate a token for a single requested period. Calendar based link provisioning was not yet part of the prototype DRAC release we used. Once the VMTC has obtained a token from the TRA, the VMTC inserts the token into the Link Access Service (LAS), at the pre-arranged time. The LAS, acting as PEP, cryptographically recognizes the token by verifying the authenticity and integrity of the provided attributes describing the lightpath using a signing method (HMAC-SHA1). If the token is valid for the requested lightpath and time period, the LAS will control an optical switch that will connect the appropriate CPU, hosting the VM, to the Amsterdam-Chicago lightpath. After the VMTC receives a positive reply from the LAS, the VMTC will initiate the move of a VM from the UvA VM site to another VM site in the setup.

4.4.4 Generic AAA toolkit components used

As indicated in the previous section, the TRA and LAS were implemented using University of Amsterdam's Java based Generic AAA toolkit. The toolkit consists of two major types of components as is further described in RFC2903 [R2903], the Generic AAA Architecture:

1. **A Rule Based Engine** (RBE), responsible for receiving, processing and replying SOAP/XML based request messages. The request can use any format including for example OASIS SAML to ensure request message authenticity and integrity. If a request message is received and parsed, the RBE will fetch a corresponding driving policy from a policy repository. The driving policy contains the logic to handle the request and calls upon one or more ASMs (see below) to

handle the meaning of a request. The policy will hand the attributes held within the request to one or more ASMs. The driving policy is a set of if-then-else statements where both the pre-condition and the *then* and *else* clauses may call ASMs that returns a Boolean value, a value of a simple type (integer) or a string.

2. **Application Specific Modules** (ASMs). ASMs understand the meaning of attributes contained within a request message. The attributes forwarded to the ASM are contained by the driving policy. ASMs may decide to use any method, including OASIS XACML, to make additional policy decisions on attributes if necessary. ASMs may forward or receive attributes from external devices or databases. An ASM may send or receive request messages to/from other RBEs, as such creating a setup of Generic AAA toolkit components. This feature enables distributed decision taking after an authorization request has been received. Both component implementations have been programmed in JAVA and deployed in a J2EE Application Server environment.

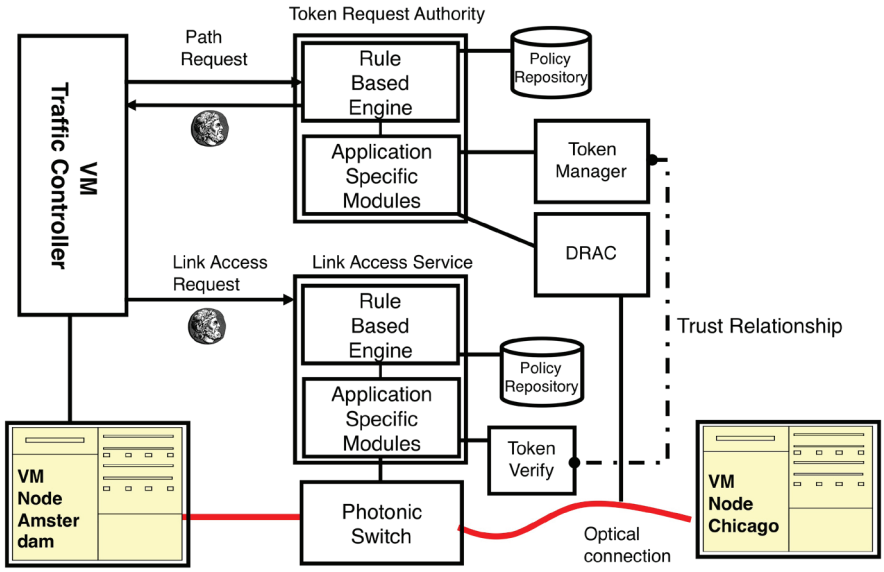


Fig 4.4.2. Generic AAA components.

For the experiment, we developed ASMs that allowed interfacing between the RBEs driving policy and both DRAC and an optical switch. We established basic XML message formats for the path authorization- and path access requests. We implemented token creation and recognition ASMs that used a key based hashing algorithm, which essentially signed the service request attributes (Token Manager) and recognized the token (Token Verify). The Token Manager will also need the hash key send via some assumed secure means (creating a trust relationship) to the Token Verify function. We wrote corresponding driving policies for the RBEs, to handle the request messages, which are send to the TRA and LAS (see Fig. 4.4.2).

4.4.5 Experiment

Our main goals for the experiment at iGrid were
- Demonstrate the described principle.
- Measure the involved timings to request a lightpath.
- Measure the time to establish a lightpath after an authorized request is received.

4.4.5.1 Demonstration of the principle

We managed to successfully demonstrate the described principle. We observed that, after a request from the VMTC was send to DRAC via the TRA, DRAC provisioned the Amsterdam-Chicago circuit and the TRA subsequently returned a token after DRAC indicated success. The VMTC initiated the token request after a corresponding operator command was issued. After the VMTC inserted a token into the LAS to move a VM, the LAS verified the validity of the token and subsequently send a control command to an optical switch in order to establish a path to access to the lightpath. Two optical switches were tested: a GlimmerGlass Intelligent Optical Switch and a Calient DiamondWave Photonic Switch. As it appeared, the Glimmerglass worked flawlessly but we found that the Calient suffered from a hardware issue causing connections to fail. These issues have been resolved now, but have prevented us to obtain data on its performance. Fig 4.4.3 shows the setup with which the measurements in this article were made. The following sections show these results. Please note that the figures obtained from DRAC represents results obtained from a prototype. This implementation does not represent the values that can be obtained from a commercial release of Nortel's DRAC product.

4.4.5.2 Obtaining a token from the TRA

Fig 4.4.3. Shows the software components involved in the measurements yielding t0 through t5 contained in table 4.4.1.

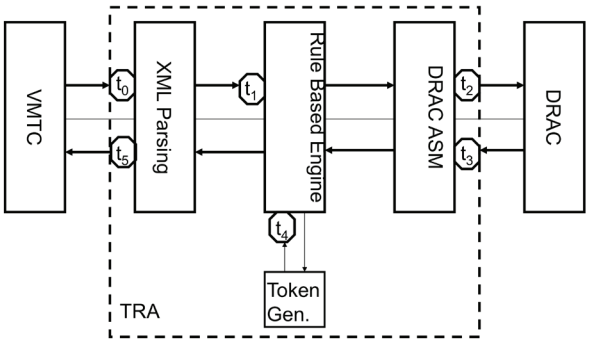


Fig 4.4.3 Token Request Authority requesting DRAC.

A token was requested by the VMTC from the TRA for a lightpath between node VanGogh5 in Amsterdam and node NUD5 in Chicago. The Token Generator generates a unique token for a specified start-time and duration after it receives a positive reply from DRAC that it provisioned the lightpath.
The TRA driving policy governs the exclusive usage of the lightpath by signing a token for each successful request made to DRAC. The policy may for example deny certain destinations to be requested at certain times. The token is not bound to the requestor, but only to service parameters (i.e. in our case Source IP, Destination IP, Start date/time and duration) so it could be exchanged or passed around by the VMTC if needed. As such, this area is open to future experiments and determines how this model can be used.
Table 4.4.1 shows the timestamps and times required to process the XML request message (t_0, t_1), the time to involve DRAC (t_2, t_3), the time to create a token (t_4) and send a reply to the VMTC (t_5). The shown data concerns a single measurement where all components ran on the same platform. We have made additional measurements and found only small differences. The most significant variation was found when measuring t_3. We found that DRACs response could sometimes be up to 10% faster than the value shown below. The other values were consistent. The token generation time could vary between 19 and 23 ms.

Time Stamp	**Event Description**	**Absolute Clock Time**	**Time-elapsed (sec.)**
t_0	AAA request received by TRA	19:25:32.776	
t_1	AAA request parsed	19:25:32:792	00.016
t_2	Request sent to DRAC	19:25:32:816	00.024
t_3	Response received from DRAC	19:25:44.056	11.240
t_4	Token Generated	19:25:44:079	00.023
t_5	Reply to VMTC	19:25:44:087	00.008
Total time elapsed (sec.)			**11.311**

Table 4.4.1. Time to open the the lightpath.

4.4.5.3 Using a token to open a lightpath.

Fig 4.4.4 shows the software components involved in measuring the time a token takes to open a lightpath using a GlimmerGlass optical switch. The PXC ASM drives the optical switch via a TL-1 style interface.

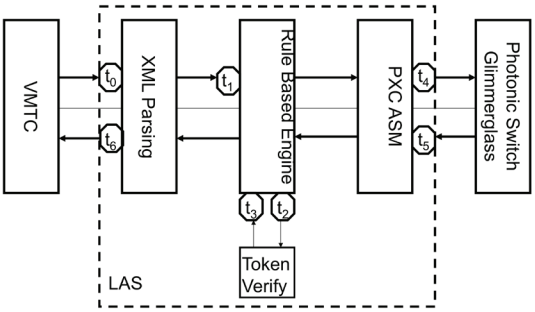


Fig 4.4.4. Link Access Service driving the Photonic Switch.

Fig 4.4.4 shows also the places where t_0 through t_6 of table 4.4.2 were measured. Fig 4.4.5 shows the relevant points were t_7 and t_8 were measured. Fig 4.4.5 also shows an abstracted view of the physical topology of the test setup. Note that we do not take the client behaviour of the VMTC into account. We start our measurements upon receiving a request by the LAS. As times were measured on components that ran on different platforms, all involved platforms (Vangogh5, the Cisco and LAS) from which we obtained time-stamps were time-synchronized using NTP.

Time Stamp	**Event Description**	**Absolute Clock Time**	**Elapsed Time (sec)**
t_0	Request received by LAS	19:25:45.278	
t_1	XML Request Parsed	19:25:45.292	00.014
t_2	Token validation start by LAS	19:25:45.306	00.014
t_3	Token validation finished by LAS	19:25:45.316	00.010
t_4	Control request send by PXC ASM to Optical Switch.	19:25:45.332	00.016
t_5	Reply received from Optical Switch	19:25:45.497	00.165
t_6	Reply returned to VMTC	19:25:45.504	00.007
t_7	Cisco 4003 switch detects light on ingress port.	19:25:53	~8 sec.
t_8	First UDP packet send by NUD5 arrives on VanGogh5 node	19:25:56.407	~3 sec.
Total time elapsed (sec.)			**11.129**

Table 4.4.2.

We have repeated our measurements a number of times, where we also included different cluster nodes and different lightpath definitions. These measurements showed no significant differences with the results shown in table 4.4.2.

4.4.5.4 Analyses

From the first measurement we can observe that the time required to process an XML request message and to generate a token (total of 71 ms) is 0.6% of the time required to have DRAC establish a path (11240 ms). The overhead caused by the Generic AAA toolkit is relatively small and independent of the complexity of the network DRAC controls.

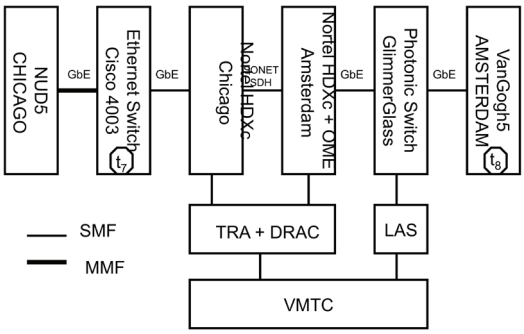


Fig 4.4.5 Timing within equipment setup.

When comparing the time required to have a driving policy handle a token validation and the optical switch to respond (total of 226 ms) with the time required to have the HDXc switches shown in fig 4.4.5 to activate the lightpath (approx. 11 sec.), the total overhead is approximately 2%. When comparing the Generic AAA toolkit overhead (61 ms) with the total lightpath activation time (11 sec), the overhead is 0.6%. We have tried to explain the 11 seconds we measured for the lightpath to become active. Hereto, before inserting the token into the LAS, we had node NUD5 in Chicago generate UDP packets using `iperf` with a `tcpdump` to monitor the interface on node VanGogh5 in Amsterdam to detect the first arrival of a packet. We also had a console window into the Cisco 4003 switch in Chicago to monitor a message generated when this switch detects a link-up state change on its interfaces. The console message generates timestamps with a 1 second resolution. Using this setup, we observed a link-up state change approximately 8 seconds after the optical switch asserted a link-up state on the ingress port of the OME / HDXc in Amsterdam. Note that the Cisco 4003 in Chicago (used to convert the Single Mode Fibre based 1000Base-LX Ethernet link to a Multi Model Fibre 1000Base-SX link) permanently asserts a link-up state on the ingress port of the Chicago HDXc. We were careful to configure the 4003 as to avoid logical link down situations caused by the execution of things like the Spanning Tree algorithm. Cisco product documentation describes possible time delays between asserting link-up state and the forwarding of the first packet. Cisco provides configuration guidelines to minimize this delay. The subsequent 3 second delay between observation of the link-up state on the 4003 and the first arrival of a packet in Amsterdam is probably a combination of the mentioned delays of the Cisco and the propagation delay of the trans-Atlantic lightpath. More experiments must substantiate the observed behaviour.

4.4.6 Conclusion

At iGrid 2005 we were able to demonstrate the token- or push-sequence principles to provide access to optical network resources. We showed that issuing and verifying tokens, using the Generic AAA toolkit, can be performed in less then 100 ms. The mechanism used XML/SOAP based messaging within a JAVA/J2EE environment. Setting up a channel within a single domain, using a high-level network abstraction provided by DRAC is in the order of 10 seconds. We observed that an optical switch causes a link-up assertion, which can take a significant amount of time to propagate via the underlying SONET/SDH based switching infrastructure. This fact may negate the advantage of using tokens in combination with lightpaths in some applications that need faster setup times. Network designs, using lightpath access switch technology that does not cause physical link up/down transitions, may avoid such delays. However, when not considering the link-up state propagation delay, a token may provide access to a lightpath in the order of 200 ms. This delay is caused by processing the request in software and the optical switch response time after receiving a switch command. When considering applications within a VM environment the use of tokens to access a pre-provisioned connection seems a promising approach. Our token-based approach needs further research to prove its efficiency towards the agent- or pull based sequences. Using for example DRAC in the agent authorization sequence would mean at least 10 seconds delay per request. However, this does not mean that such an approach is inefficient. The efficiency will also depend on the applications lightpath usage profile. Also comparing the Glimmerglass timing with timings obtained from a Calient optical switch would further substantiate our findings.
Once obtained by the VMTC, one way forward could be that tokens are passed up to a VM application or to a VM scheduler or monitor. Such monitor could for example determine a site usage pattern and, based on both the usage and guaranteed lightpath availability, decide to move a VM. An intelligent scheduler could negotiate an optimal timeslot for a certain VM to move. If the lightpath is not needed, the token could be placed in a pool and make it available to other applications. Another way forward is to consider the management of the driving policy of AAA components that request, issue, distribute, use and accept tokens. We are creating web-based mechanisms that allow both resource owners and users to manage appropriate parts of the driving policy. As such, we expect to create scenarios where stakeholders are allowed more flexible and intuitive control of their policies.

4.5 Token sequence authorization applied to network cases[11]

Section 3.4.3. first summarised the various concepts demonstrated during the period 2005-2007 and shows the evolution of the token based concept into something that got named the "Token Validation Service" that helped to implement a multi-domain scenario. The concept was demonstrated during SuperComputing 2007 in collaboration with Nortel and Internet2.

---

11 This section is based on the experimental part of publication:
"Multi-Domain Lightpath Authorization using Tokens"Leon Gommans, Li Xu, Fred Wan, Yuri Demchenko, Mihai Cristea, Robert Meijer, Cees de Laat , , Future Generation Computing Systems, Vol 25, issue 2, 2008, pp 153-160, DOI 10.1016/j.future.2008.07.013

4.5.1 Implementation of the token based access control method

Token based access control mechanisms have been implemented as components of a general authorisation infrastructure for network resource provisioning. It is used to simplify access control to reserved distributed resources in a multi-domain environment. The infrastructure, called the Generic AAA toolkit (GAAA-TK) [AAATK], is being developed by the University of Amsterdam (UvA). The toolkit both implements a number of security mechanisms to support the multi-domain policy based authorisation process, as well as token-based access control. As such, the Token Validation Service (TVS) has been developed as a special component to support token handling at all stages of the general network resource provisioning. It supports inter-domain token based signalling during the reservation stage. It performs path and reservation context distribution at the provisioning stage. It also provides the token validation service at the access phase. The GAAA-TK is provided as a pluggable Java library and as a standalone domain central authorisation service (DCAS). The special GAAA-TK profile and TVS implementation includes support for all layers mentioned in section 3.4.4 The GAAA-TK also implements the SAML-XACML [DEM7] authorisation request–response protocol that allows for authorisation request evaluation with the local or remote XACML based Policy Decision Point (PDP).

Although the TVS component has been implemented as a part of the general GAAA-TK library, it can also be used separately. All basic TVS functions are accessible and requested via a Java API. As such it can be used with other authorisation services implementations and frameworks such as Globus Toolkit Authorisation Framework [GTAF] and PERMIS [PERM] to support necessary functionality for token distribution and processing in their target application areas. Further TVS development will extend Web Services interface to allow all TVS functions be accessible via Web services. The current TVS implementation supports both shared secret and PKI based token key distribution.

4.5.2 Demonstration of the token principle

In this section we will present some of the work that has been done within the context of projects that collaborate and share information directly or indirectly with the OptIPuter project. Various aspects of the tree and chain token approach where demonstrated at different occasions.

4.5.2.1 The packet level approach

The packet level approach was demonstrated using an Intel IXDP 2850 NPU development platform programmed as Token Based Switch (TBS) at SC2005 [GOM5]. Here a token, inserted into the IP Options field, enabled IP packets to take a specific pre-provisioned lightpath. This offers IP layer support at stage 3. For implementation details we refer to [CRIS]. OGF document GFD.083 Firewall Issues Overview [GFD83] argues that this kind of switch could form a potential solution for a firewall, protecting hybrid network resources if public access needs to be supported as mentioned in Section 3.4.2.1. In later releases of the TBS we programmed it to forward a

token received, at IP layer, to the RSVP-TE layer and also XML/SOAP layer, as such acting as a token gateway.

4.5.2.2 The path signalling approach

This approach was the subject of our demonstrations during SC2006. Fig. 4.5.1 illustrates its components.

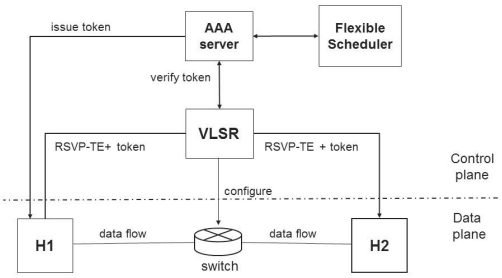


Fig .4.5.1 Token-based GMPLS at the path layer.

Here we demonstrated how a GMPLS based network is able to support tokens by including it in a specific field of a RSVP-TE PATH signalling message. To show this ability we modified the Virtual Label Switch Router (VLSR) and Client System Agent (CSA) code of the open source GMPLS project—DRAGON [LEHM] to recognise tokens. In this demo, the mentioned elastic scheduler (section 3.4.3.3) acted as an advance reservation resource manager to take decisions for stage 1. The token was stored inside the AAA server for stage 2. At stage 3, the tokens were inserted into a Policy_Data object (RFC2750) of RSVP-TE PATH messages that are exchanged between hosts and the VLSR to signal the data-path setup. The VLSR parses the request message and verifies the token by querying the Generic AAA server. If the token is signalled as valid, the VLSR forwards the message to the next hop and configures the switch in the data plane.

4.5.2.3 The service layer signalling approach

This approach was subject of a single domain demonstration during iGrid 2005 and Supercomputing 2005 and a multi-domain case during SuperComputing 2007.

Single domain case: The VM migration experiment.

In our Supercomputing and iGrid experiments in 2005, we used a Generic AAA server from the GAAA-TK as Policy Decision Point (PDP) [R2748]. The Generic AAA server architecture is described in more detail by RFC2903 [R2903].

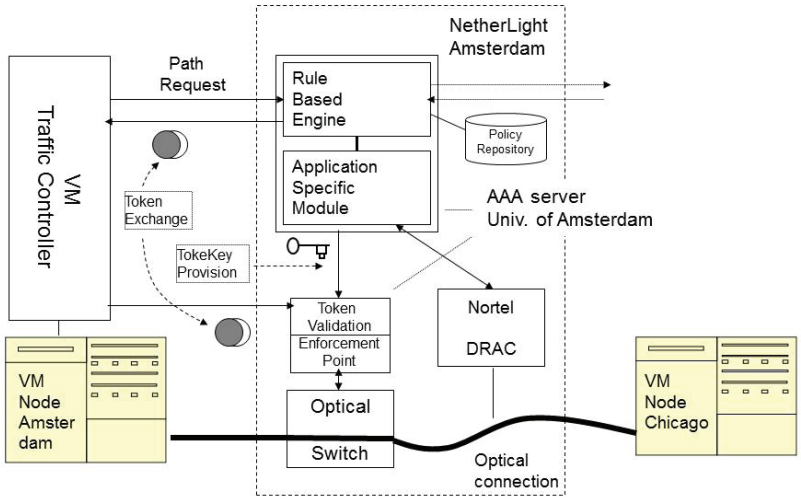


Fig 4.5.2. The VM Migration experiment. (see fig 4.4.2 for larger version)

Fig. 4.5.2 shows the basic setup of the experiment performed during iGrid 2005 and SC2005. The experiments [GOM62, TRAV] were conducted in collaboration with Nortel. Here we showed the migration of a XEN Virtual Machine (VM) across a lightpath. A Micro Electro-Mechanical Systems (MEMS) based Optical Switch enforced access to the lightpath by switching an authorized CPU of a cluster to the designated lightpath. Nortel's Dynamic Resource Allocation Controller (DRAC) was in control of provisioning and resource management of a lightpath between Amsterdam and Chicago.

In the demo-scenario, a VM Traffic controller migrated a Virtual Machine via a given lightpath initiated at stage 1. After contacting the DRAC to check if the request can be honoured, the Generic AAA server (consisting of a Rule Based Engine and Application Specific Modules—see RFC2903) generated a token. During stage 2, the Generic AAA server provisioned the Token Validation (Policy) Enforcement Point with the token-key. At the appropriate time, i.e. at stage 3 when the actual migration of the VM is about to happen, the VM Traffic controller will insert the token into the Token Enforcement Point. If the token is accepted, this function will control the Optical Switch, such that it will connect the right VM node to the right optical path. The DRAC was assumed to provision the circuit at the agreed time. The mechanism will prevent different VMs from migrating at the same time using the same resource. This example shows that applications can be more accurately associated with a lightpath as stated in Section 3.4.2.

A multi-domain case: Implementation of the Token Validation Service.

At Supercomputing 2007 we proposed, and implemented, the token concept into the IDC control plane of Internet2 DCN. The Token Validation Service (TVS) mentioned earlier was integrated

with the IDC, as shown in Fig. 4.5.3. The TVS enables an IDC to generate and communicate tokens much in the same way as illustrated in Fig. 4.3.2. In the above example a reservation application obtains a token from the chain of IDCs in the same way as described in 3.4.3.1. In the scenario we developed for SC2007, a token was subsequently placed on a USB memory stick and carried to a MacMini with a Full HD-TV display to show a movie streamed from a CineGrid [CINE] server in Amsterdam via a 1 Gbps DCN link. The difference with the previous examples is that Internet2 implemented an IDC version where tokens were handed back to the Lightpath Authority at stage 3. This was considered the easiest solution for a first implementation. The path signalling way, using a LightPath Service implemented with the GMPLS implementation from the DRAGON [DRAG] project together with a Policy Enforcement Point developed as part of the TVS at UvA, was implemented at a UvA testbed. Also, not all domains may want to support token enforcement. Fig. 4.5.3 shows domain A without such ability. For such cases, the inter IDC protocol supported transparent pass-through of tokens. After a reservation is made, and the enforcement points are provisioned (stage 1 and 2 complete), the IDC is signalled to open the reserved path for stage 3 using only SOAP/XML messages.

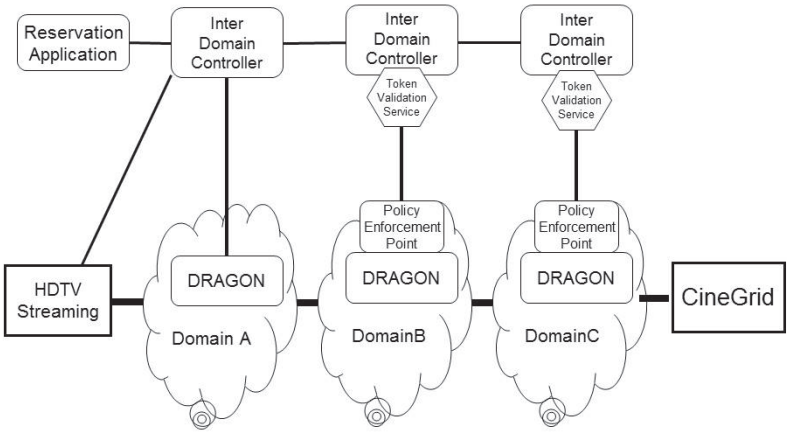


Fig. 4.5.3. The Token Validation Service Experiment at SC2007.

4.5.3 Future work

Combining the use of tokens with the tree and chain signalling approach, that use the same interface for a domain will be a key item for further research. This should enable domains to authorise the use of network resources to create a lightpath in many different scenarios. Also some form of GRI and token format will need to be agreed upon, such that domains can identify lightpath sessions, and base their internal administration and enforcement on it. Work on this within the GLIF and research projects, such as Phosphorus and GigaPort, is currently ongoing. In a simple scenario, the TVS can be programmed with a shared key, using a Web Services

interface to facilitate communication between the Lightpath Authority and Lightpath Service. A more flexible and automatic TVS security model may use an Identity Based Cryptography [SHAM] (IBC) approach that relies on a domains IBC key generation service.

4.5.4 Conclusions

The paper presented the results of our ongoing research and development to build a consistent authorisation architecture and flexible access control infrastructure for multi-domain hybrid network provisioning. The proposed and discussed concepts and solutions use a common abstract token concept.

We have shown that a token can act as s*hared abstract permission that is presented as part of an access request in each domain* where its permission is represented as an index pointing at a pre-allocated network resource. In multi-domain scenarios, the same token may point to different definitions of a lightpath segment inside different domains. As such a token can be considered as glue to collect authorisations to use network segments inside different domains, forming an end-to-end lightpath.

We showed how tokens are used at all three stages of the RFC2904 based resource provisioning sequence: The access token is created as a result of the successful phase 1, during which the multi-domain path is reserved. During the following phase (2) the reservation and token context information (including the token key) is provisioned to all participating domains. In the following lightpath access phase (3) the token is used to enforce access to the network resource. The abstract nature and small size of tokens allow their use for access control enforcement at three different networking layers: the IP layer, the path layer and the service layer and showed examples of their usage. We also showed that two different models are common during the collection of authorisations to create a token: the tree and chain model.

In our experiments and demonstrators we proved that the token mechanism is a flexible and powerful way to allow different domains to share and enforce lightpath authorisations.

We exploited simplicity and flexibility of the token as it can be contained by different protocols, and is able to be passed on between protocols. The GMPLS control plane can forward the token inside XML based messages such as SAML assertions. Also the fact that the usage of the token is completely independent from the way domains negotiate in either the tree or chain fashion is a powerful concept that facilitates interoperability. A lightpath service that enforces tokens does not care how it receives the provision information at stage 2 as described in Section 3.4.2.4. Further investigation of these characteristics is a logical continuation path for our research into how domains can interact to offer authorised lightpath services. We proposed and jointly used the GRI concept as a common session identifier in our collaborative effort with the Internet2 DCN project. The GRI was used as a resource identifier that is created at the beginning/start of the provisioning session/process to simplify the provisioning process tracking. We looked at a signed GRI as a possible form of a token. We found that this format enables each domain to keep administrative details of its lightpath segment hidden from other domains, whilst referring to the same end-to-end path. The token subsequently allows domains to enforce access to its resources without the need for an unpredictable overhead to contact the authority. As such, tokens offer a

fast and flexible way to allow different domains to share and enforce lightpath authorisations. In our SC2007 demo we consolidated the token concepts into a Token Validation Service (TVS). The TVS supports token handling at all stages of the general network resource provisioning sequence.

4.6 Summary

It this chapter we showed experiments that have been performed with the concepts of the Generic AAA Architecture and Authorization Framework at different levels of network technology and at different places in the network. We demonstrated the ability of driving policies to control network configurations by using the functional elements of the Generic AAA architecture.
By measuring the timings involved, we demonstrated the limitations of the Agent model in a multi-domain setup. These experiments made us re-consider the applicability of the Agent model and started to look at token models that inherently separates the request for a network path from the use of a network path. We demonstrated that a token is a simple but powerful concept that can be implemented at different network layers. The token model is more suitable for multi-domain scenario's then a pure Agent model based scenario due to the latencies involved in processing a request.

Organising trust in multidomain scenario's

5

"It is clearly better that property should be private, but the use of it common; and the special business of the legislator is to create in men this benevolent disposition."

Aristotle (384 B.C. - 322 B.C.)
Greek philosopher

5 Organising trust in multi-domain scenario's [12]

In RFC2904 [R2904] we noted that: *trust relationships are necessary for authorization transactions to take place.* In this chapter we will elaborate on the concept of trust. We describe a framework helping us think about ways how trust can be organized in scenario's where entities provide a service to a user as a group. It is important to recognize that in such group, each member cannot provide a service on its own and therefore must collaborate with other members. For example: Banks cannot provide worldwide credit card services on their own and must therefore collaborate with other banks to allow worldwide payment services to be provided to its customers (merchants and cardholders).

The work on the Service Provider Group (SPG) Framework examines what trust really means in a multi-domain service provider context. It considers how trust can operationalized in policies executed in each domain in such a way that other domains will trust its execution. To understand this kind of trust, we examined already existing forms of collaborating organisations that are competitors, but also see benefit in collaborating such as with providing credit card services. From that study we extracted a framework that is helpful when conceptualizing new situations. As such the framework will describe the Network Provider Group as a specific incarnation of the SPG Framework. It is important to recognize that in multi-domain cases each member should be able to maintain its autonomy and is only willing to give up some of its autonomy when members see a clear overall benefit.

Both within the Business and e-Science world, the use of virtualized resources is growing rapidly. These resources are increasingly delivered by multiple converged infrastructures, e.g. clouds that combine server, storage, and network resources from different providers. Such development requires careful re-thinking of the trust framework used between providers. As the scale and complexity of virtualization grows, so does the complexity of authorizing resource chains that are arranged across multiple providers. This type of authorization requires pre-establishment of trust relationships between providers and arranging some level of power. The paper presented in this chapter studies the roles of trust and power when considering the requirements of authorization protocol exchanges between providers. Establishing power in the form of impersonal rules is a key element to conduce the necessary trust between providers. The Service Provider Group (SPG) is a way to arrange such power. The SPG framework provides a way to organize thinking about multi-provider services and can be used to describe emerging collaborations such as those found within the realm of optical network service provisioning.

---

12 This section has been based on publication:
"The Service Provider Group Framework", Leon Gommans, John Vollbrecht, Betty Gommans-de Bruijn, Cees de Laat, Future Generation Computer Systems. DOI: 10.1016/j.future.2014.06.002

5.1 Introduction

Increasingly, automated mechanisms are used that exchange protocol messages arranging, authorizing and provisioning end-to-end chains of compute, storage and network elements as a service. Delivery of end-to-end services, not only in e-Infrastructures, needs coordination and oversight to ensure quality, manage risk and possibly liability. Users typically do not want to carry the burden of such coordination and oversight. The ability to arrange end-to-end services by a group of providers reliably (with adequate coordination, oversight and transparency in accordance with the terms and conditions of service agreements) influences the willingness of both users and service providers to rely on each other. Willingness to rely on something or somebody is an important understanding that is associated with trust.

To avoid damaging trust of users vested in an offered service, it is important that each provider in the chain shares a common, well-defined understanding of the terms and implications of a service agreement when authorizing the use of its contribution. Trust is needed to define service agreements that are embedded in a commonly understood set of rules. Power is often needed to enforce its terms and implications. When a group of service providers come together and recognize the benefit of collaborating[13], such an agreement is typically based on each participant personally trusting one another. Power, for example enforcement of written group admission rules[14], is used to ensure a participant can be trusted to contribute according to the spirit of the group.

As the number of participants in a group increases, the level of automation increases and the services are being increasingly relied upon, the concepts of personal trust and power will inherently become more impersonal. Establishing a Service Provider Group (SPG) is one way to arrange *impersonal power* (rules) such that it conduces trust amongst group members. Instituting a SPG is a way to establish and maintain a common set of inter-organizational rules that are translated into intra-organizational policies such that each entity *knows that the policy it is authorizing is correct*. We make the assumption that protocols, exchanging authorization transactions between organizations, will provide enough message confidentiality, authenticity and integrity such that the security of an exchange is never disputed.

We consider a SPG as a group of member organizations that act together as a business. A SPG provides one or more services that none of its members could provide on their own. To a user, the SPG appears as a single provider. To members the SPG appears as a collaborative group with standards and rules that each member translates into conforming policies. The policies regulate the provisioning of services and the user terms and conditions that are enforced by the group. A user signs a service agreement with a member representing the SPG. Members may or may not have users or may or may not provide services as a contribution to the group. A member has signed a

13 An example is GLIF, that was established by 33 participants at the 3rd annual Global LambdaGrid Workshop, held August 27, 2003 in Reykjavik, Iceland

14 For example: GLIF is open to any owner/custodian of lambda infrastructure (lightpaths, exchange points, etc) that is willing and able to make that infrastructure available to other GLIF participants on an agreed basis when it is not required for its own needs. (Source GLIF Strawmen Charter)

membership agreement with the group. The SPG has some sort of directorate role that oversees the interactions and interoperation of its members. Fig. 5.1 shows the basic elements of a SPG. The paper will focus on the "human managed" business part of the SPG resulting in policies that are capable of determining the operational part of the service provisioning that is typically "protocol managed".

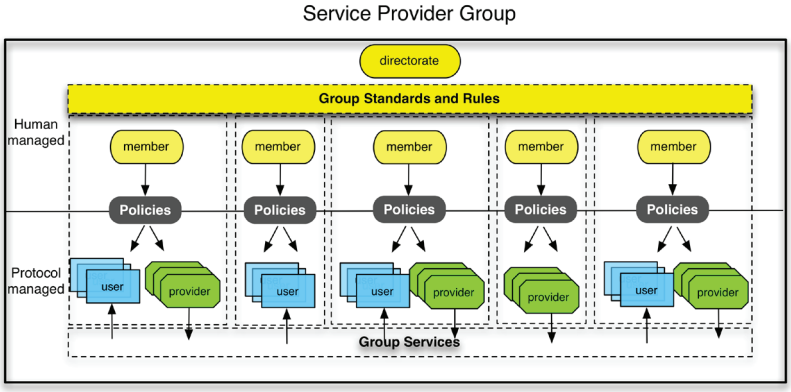


Fig 5.1. The Service Provider Group Framework basic elements.

The framework builds on RFC2904 (AAA Authorization Framework [R2904]), which recognized that rules must be in place before authorization transactions can take place. Fig. 5.1 shows that SPG group rules and standards are defined at business level involving human members that translate them into policies. Policies are executed using protocols by elements that provide services. Policies also govern user interactions obtaining group services.

In section 5.2, we start with a study of what trust and power means from the area of organizational sciences and see how the SPG can be positioned herein. We then consider in section 5.3 the rules of a mature example taken from the Payment Card Industry and put it into the trust and power context. This leads us in section 5.4 to a framework containing the essential elements of a SPG detailed in section 5.5.

This study has been motivated by the fact that members of MasterCard together handle payment transaction authorizations as a collaborating group of financial service providers. As such, we argue that this collaboration can be seen as a successful example of a SPG. The paper abstracts a framework for a SPG from observing the rules MasterCard uses to establish trust and power that are subsequently transformed into policies governing interactions between users and members at operational level. SPG members interoperate with each other using policies and protocols that are monitored and enforced. An existing networking example, eduroam providing WiFi access to students worldwide, is used to verify observations made to establish the framework.

To better understand the relationships between trust, power, rules, administration and enforcement of policies, etc. concepts from organizational science are considered to explain them. It will recognize why impersonal power is a means to conduce trust efficiently within

and between organizations. We then propose a framework that recognizes three types of power that loosely resembles the concept of the Trias Politica. We will show how these powers are used to administer and control the functional levels of organizations. This is done by means of policies that are provisioned and enforced such that each participating organization is able to rely on the fact that *policies are known to be correct*. Being able to rely on such knowledge has important consequences for the protocol(s) used to communicate authorization decisions. Precise knowledge and power to enforce authorization decisions allow the semantics of such decisions to be abstracted as much as possible when being communicated across entities. When applied to connection oriented networking, a decision that authorizes an end-to-end connection, for example represented by a token, may carry different detailed meanings within each provider domain. However the effective outcome of a decision must be the same for every domain to create a uniform SPG defined network service. For example, some providers may decide as a policy to always carry connections across high-available circuits, using redundancy, even if the service request specifies a best effort service that do not require redundancy. Another provider will route such connections without redundancy. Both providers know from the common rules that their policies are both correct when providing a service.

The paper also introduces briefly the concept of a Network Provider Group (NPG) that provides network connections across multiple domains. The NPG concept, an e-Infrastructure style incarnation of the SPG, is intended to allow independent network connection providers to interoperate to provide connections to its users as a group defined service.

Lastly, the paper will argue that well-defined rules can help multi-domain agent scenario's, where tokens based sequences are used to authorize services, is able to minimize the amount of information that needs to be exchanged in protocol objects.

5.1.1 Related work and motivation

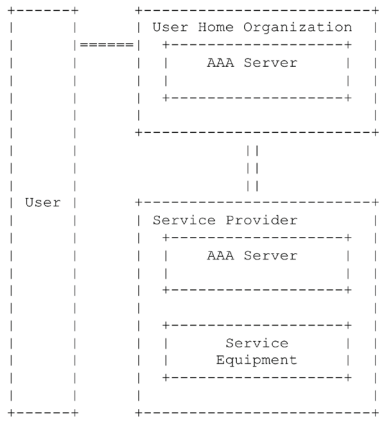


Fig 5.2 Service Agreements

In 2000 the IRTF Authentication, Authorization and Accounting (AAA) Architecture Research Group published RFC2904 [R2904], describing a framework for authorization. The document identifies three basic conceptual entities involved in an authorization transaction. The AAA Server: Capable of making policy based authorization decisions, such as generically described in RFC2903 [R2903]. The User: Making a service request that needs to be authorized. The Service Equipment: A resource in need of knowing if a User request can be granted based on the execution of some policy. The RFC organizes these entities into a Service Provider that owns

Service Equipment and a User Home Organization, registering details of users involved in the authorization decision.

Fig. 5.2 shows these elements. The diagram includes double lines that represent Service Agreements that must pre-exist in some form between the organizational elements. When multiple User Home Organizations and Service Providers start to collaborate, arranging service agreements becomes more complex and is likely to become a role of some form of organization. As a new contribution, the SPG framework provides a way to think about dealing with this complexity.

Service Agreements are also called Trust Relationships in some of the AAA application examples [R2905] presented in RFC2905. Trust Relationships and Service Agreements are intertwined concepts where one cannot live without the other. Trust Relationships can be built at protocol level as result of an established Service Agreement or a Service Agreement can be seen as embedded in a Trusted Relationship between business actors and can as such be seen as a result of such relationship. We will focus on the latter notion that is more related to the human managed business level. The first notion is related to the operational/protocol level as discussed in RFC2904. It describes a number of typical sequences by which these entities may communicate to handle authorization requests (Push-, Pull- and Agent sequence). Protocols languages, including RADIUS, DIAMETER, COPS, SAML, etc., can be used to implement such sequences. Within the Networking domain, research has gone into exploring suitable protocols and mechanisms (e.g. GLIF [GLIF], Internet2 ION [ION], ESNet SDN [ESN], GÉANT Autobahn [GEA] and G-Lambda [LAMB]). This research is aimed at the automated creation of dedicated bandwidth network connections across multiple autonomous Service Provider Networks using one, or a combination of these typical sequences. The telecommunications industry performs research and standardization in for example in the Telecommunication Management Forum Frameworx effort [TMF].

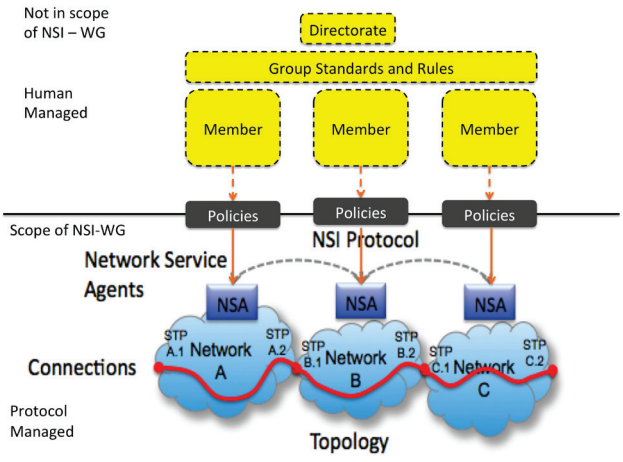


Fig 5.3: NSA's create automatically network connections by chaining network services from different networks. Arranging group standards and rules for collaborating members that determine policy elements that are known to be correct is not in scope of the NSI-WG effort

Recent activities in the Open Grid Forum (OGF) such as the OCCI [OCCI] working group and Network Services Interface [NSI] working group (NSI-WG) have been fruitful in standardizing the interworking between service domains. The OCCI working group establishes an API and protocols for management of Cloud resources. Within the NSI work [GFD173], as shown as example in fig. 5.3, Network Service Agents (NSA's) are capable of coordinating connection establishment across multiple networks in one of many possible topologies using the NSI protocol. It is however not in scope of the NSI effort to arrange a common set of rules and standards, which determine the offered service and provides trust amongst collaborating providers acting as member of a group. The material presented here may contribute by further describing the dashed elements shown as "Human managed" in fig 5.3.

Before NSA's can participate in authorization transactions, which would allow such connections to be created, RFC2904 recognizes that:

"There must be a set of known rules in place between entities in order for authorization transactions to be executed. Trust is necessary to allow each entity to "know" that the policy it is authorizing is correct. This is a business issue as well as a protocol issue."

The above statement touches on the concept of trust governing authorization transactions. It is essential to understand that each entity participating in the handling of a transaction must have trust in the fact that the policy - as a means to perform an authorization - is the correct one. This fact creates a system where each entity knows to behave correctly in such a way that:

Trust Notion 1: Users trust the predictability of the system's outcome as a whole.
Trust Notion 2: Collaborating entities trust each other to act in a correct, coordinated and predictable way, e.g. Service Providers and User Home Organizations in the RFC2904 case trust each other as members of a collaborating group.

As does RFC2904, the mentioned research project initiatives assume trust relationships - based on service agreements and known rules - to pre-exist before authorization transactions can take place. In order to augment the mentioned research and also the standardization efforts such as the NSI work, the material of this chapter contributes to the research by focusing on the business issue side of the RFC2904 assumption.

We will show that the concept of a SPG is a viable option to arrange in particular Trust Notion 2: The case where an increasing group of providers need to collaborate as members of such a group to authorize and deliver an economic service. We have assumed that fulfilling Trust Notion 2 will be the basis to earn the Trust of Notion 1 (Users trusting the predictable outcome of a system).
To further explain the need for an SPG Framework we will now consider needs both identified in general and also specifically for the optical networking case. The specific case we have called the Network Provider Group case. We will then identify some requirements for a SPG.

5.1.2 The need for a Service Provider Group Framework

5.1.2.1 Expressed needs for a Service Provider Group Framework

When studying the Anatomy of the Grid [FOST] in 2001, Ian Foster et. al. recognized that:

"The real and specific problem that underlies the Grid concept is coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. The sharing that we are concerned with is not primarily file exchange but rather direct access to computers, software, data, and other resources, as is required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and engineering. This sharing is, necessarily, highly controlled, with resource providers and consumers defining clearly and carefully just what is shared, who is allowed to share, and the conditions under which sharing occurs. A set of individuals and/or institutions defined by such sharing rules form what we call a virtual organization (VO)."

Ian Foster recognizes the need for a set of individuals and/or institutions that define sharing rules and rules that allow coordinated resource access and usage. Allowing VO's to define and implement sharing rules has been implemented by initiatives like EGI [EGI] that help National Grid Initiatives provide Grid Services across Europe to researchers and is successfully evolving.

In his position paper [NEGG], Kees Neggers (at that time Director of SURFnet, the National Research & Education Network of the Netherlands and a founding participant of the GLIF), recommended in 2011:

"European and national investments should together lead to a global service concept. This concept should be based on a federation of networking resources and technologies owned and operated by a variety of national, regional, European and international partners, coordinated in Europe through a collaborative effort under the GÉANT label".

Neggers envisaged a global service concept as an organization, coordinating the delivery of a service under a single label. This concept should arrange the interactions between technologies owned and operated by a federation of autonomous partners.

Within recent cloud developments we can observe that:

1. The NIST Cloud Reference Architecture [FLIU] recognizes the functions of a Cloud Broker and Cloud Auditor that are defined as:

- Cloud Auditor: *A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation.*
- Cloud Broker: *An entity that manages the use, performance and delivery of cloud services, and negotiates relationships between Cloud Providers and Cloud Consumers.*

2. Cloud Infrastructure as a Service (IaaS) implementations are increasingly considering open source based cloud provisioning distributions like Open Nebula and OpenStack using an open

standards based management interface like OGF's OCCI [OCCI] allowing multiple external and internal cloud offerings to be *integrated*.

3. Gartner [GART] defines hybrid cloud computing as: *"policy-based and coordinated service provisioning, use and management across a mixture of internal and external cloud services."*

Functions of entities such as brokerage, policy based and coordinated service delivery, performing assessments, establishing and negotiating relationships, declaring the use of open standards for integration are essential to allow complex cloud scenario's to be established.

We took these general observations as evidence that there is a need in the evolving e-world to consider the development of a SPG framework describing elements that provides rules and policies coordinating resource access and usage. It should also describe functions such as monitoring of services and its availability, performance and security, negotiation of relationships, integration of service offerings, etc. An organization based on the SPG Framework could be made responsible for performing such functions based on a common set of rules.

5.1.2.2 Specific Need: The Network Provider Group

The Network Provider Group (NPG) concept, as a specific incarnation of the SPG concept, is motivated by the desire to understand how networks from different geographic areas can interact with each other to provide high bandwidth dynamic connections. National and regional optical networks have been created in different parts of the world (GLIF map [GLMA]). Exchange points have been built where these networks interconnect (Dijkstra & de Laat [DIJKG]). Protocols for automating interconnection, developed by the NSI work group, are deployed (Chin P. Guok [GUOK]). Some applications have been developed that take advantage of existing capabilities, and the potential to serve many more applications exists if the connection creation process can be automated (Internet2, 2012 [IN2I]) (Géant, 2012 [GEASC]).

The concern in a NPG is how resources used in such connections can be allocated with confidence to specific users. This requires resource providers to know resource requestors, and have some way of deciding whether to give a resource to a particular user at a particular time. We will show how confidence can be built into the automating protocols.

The policy part of a NPG determines how collaborating members interact to define the service to be offered and how it is monitored and enforced. The operation part defines the protocol used to create and measure connections. At business level, the trust users have in NPG services as a whole, is trust as meant by Trust Notion 1 of section 5.1.1. This trust is based on Trust Notion 2: The trust that deals with assurance that transactions are performed correctly at operational level by the organizations that are supposed to perform them.

NPGs are needed so that users can be sure they get the connection they request and that providers can be sure they are providing the correct service to a known user. Within the group, each individual provider maintains autonomy and ability to authorize its part of the connection using its own policies that are based on group rules.

5.1.3 SPG Framework Requirements

The framework must describe how service providers, typically providing competitive services to the same market domain, could be structured to set up a collaboration that creates a chain of individually provided services. To deliver such services, the establishment of a SPG only makes sense when it provides a benefit to all of its members. The user expects the SPG to arrange a consistent service quality across multiple providers once service access has been authorized. After the SPG is put into existence and new members join, each new member must be offered efficient ways to acquire knowledge on how to correctly add its part to commonly defined services. Whilst taking its own policies and possibly National Law into account, a new member should be (within reason) able to correctly implement SPG defined rules. Members must develop policies that are used for authorization, coordination and delivery of a group service and to manage and avoid possible consequences of deviations. In essence common SPG rules and derived policies must be administered and enforced with each participating domain in such a way that all service providers act as one to allow benefit for all.

As MasterCard managed to achieve such service delivery across the globe, we can examine how its power creates trust amongst its members and users and consider how such power is implemented. Before we can do this, we must first understand the meaning of the concepts trust and power and how these concepts can be positioned in our context.

5.2 What do we mean by Trust?

Trust is a broad concept studied in areas such as sociology and psychology. We have placed trust in the organizational context. Here, different actors (persons or organizations) need to have relationships that coordinate business activities that deliver goods or services. An elaborate overview of the concepts used within this context can be found in studies performed by Nootenboom [NOO3] and Bachmann [BAC1]. The following descriptions have been extracted from their studies and are used as a definition:

Trust in the organizational context is predominantly considered as a *means to cope with uncertainty*. Trust reduces uncertainty by allowing specific (rather than arbitrary) assumptions to be made about an actor's future behaviour. Trust inherently introduces a risk as trust can be disappointed. Finding *good reasons* can minimize such risk. Note that if the risk of disappointment does not exist, then trust is not needed. Regulation and its *potential of sanctioning* is an effective remedy to confine risk by providing *good reasons*. When sanctioning is used however, it destroys trust and should

therefore be used with care and reason. Commercial law and contracting practices are important elements that embed trans-organizational business relationships such that actors know what is expected from a good relationship.

Trust can be tacit, i.e. be an understanding living in the mind of one or more persons. Such understanding can be made explicit, i.e. written down and as such be impersonalized. In this form it can act as a set of rules people or organizations can live by. When the rules are enforced, rules become the base of power. Let us look at how both forms (the personal and impersonal form) are distinguished in an attempt to position the role of a SPG.

5.2.1 Personal and Impersonal Trust

Bachmann [BAC3] specifically distinguishes between *"Personal Trust"* and *"Impersonal Trust"*. Personal trust is trust that is formed by interaction between persons and grows with experience between one person and another. *Impersonal trust* is trust that is rooted in the tacit understanding of personal trust, which is subsequently externalized and expanded into an explicit form of knowledge captured in law, rules, codes of conduct, etc.

With *impersonal trust* Bachman further distinguishes *"System Trust"* as trust *in the object* of trust and *"Institutional trust"* as trust *in the relationship* between actors that is embedded in the institutional framework.

System trust can be seen as *confidence* in rules and involved authorities executing such rules. A good example is the aviation system where ICAO, FAA, EASA and other (national) aviation regulatory bodies and authorities ensure the safety of citizens boarding a commercial plane. Citizens do not have to be aware of the risk involved in commercial flying but instead *have trust in the aviation system*. Important part of the system are the rules and regulations governing authorities that administer, qualify and oversee airlines, aircraft manufacturers, flight training organizations, maintenance organizations, etc. Authorities are given the power to qualify, licence and monitor organizations and enforce rules, regulations, procedures, standards, etc. regarding the quality of aircrafts, its maintenance, operation, safety procedures, training, etc.

Institutional trust is trust in the relationship between actors embedded in legitimized normative rules, codes of conduct, standards, etc. Such rules, conduct, standards are legitimized by for example trade organizations, industry forums, professional associations, standards bodies, etc. Institutional trust plays for example an important role in sport, where participants are expected to compete in accordance to a set of rules established by an international sports union or federation. In competitions, participants trust each other that nobody cheats e.g. by taking drugs. Such trust is embedded in the impersonal social rules rooted in personal understanding and institutionalized by organizations such as WADA [WADA].

A glossary of terms used until now are summarized in table 5.1.

Term	Description
Trust *(within the organizational context)*	A means to cope with uncertainty
The risk of trust	Trust inherently introduces risk as trust can be disappointed.
Good reasons	Good reasons can be used to minimize risk.
Sanctioning	The potential of sanctioning is an effective remedy to confine risk by providing good reasons. When sanctioning is used it destroys trust.
Personal Trust	Experiences individuals make with each other in the course of frequent interaction over a longer period of time
(Impersonal) System Trust	Trust an individual has in the functioning and *in* the reliability of impersonal social structures
(Impersonal) Institutional Trust	Trust *between* individuals vis-à-vis existing impersonal social rules
Social Structure *(at individual level)*	The way norms shape the behaviour of actors within the social system.
Institutional Framework	The systems of formal laws, regulations, and procedures, and informal conventions, customs, and norms, that shape socioeconomic activity and behaviour.

Table 5.1: Trust concepts within the organizational context.

Bachmann notes that powerful institutional rules are able to control the expected behaviour that can both absorb risk and increase the chance that trust becomes a preferred mechanism to control the expected behaviour of actors. Unfortunately such trust is sometimes misplaced (as seen in several recent cases in the world of pro cycling). Trust is therefore not absolute and power is there to assist in an attempt to prevent misplacement. When power is used, for example by revoking titles and medals, trust is damaged.

This leads to the question of what the role of power is in relation to the concept of trust.

5.2.2 The role of Power

Trust is not the only mechanism that can be used to make assumptions about an actor's expected behaviour. Power is a similar mechanism to trust as it too influences the selection of actions of an actor in the face of consequences. Power and trust are both means to achieve the same goal of coping with uncertainty. In essence trust makes *positive* assumptions about the willingness and ability of an actor to co-operate, whilst power is based on *negative* assumptions implying consequences. In practice, most relationships are based on a mixture of trust and power, where

one of the mechanisms has a predominant role. Trust is often preferred as predominant role. However, when the *impact of risk* plays a significant role, relationships tend to rely more on power. Power will however only work if there is a realistic threat of sanctions.
Both trust and power can be applicable in personal and impersonal sense. In a parent-child relationship a child personally trusts a parent. Based on experience, a parent may have good reasons to personally trust the child to always attend school. If this trust is misplaced, a parent can use its personal power to ensure that a child will face consequences if its behaviour continues. The impersonal (institutional) power of the school and (system) power of the authorities may help parents in providing good reasons (consequences) to both child and parent. School (impersonally) and parents (personally) trust children not to cheat when taking a test by providing good reasons in the sense that both will stress that learning is important for the child's future and that cheating is a socially unaccepted behaviour. Good reasons are embedded in the social framework of a child. In addition a child will experience the impersonal power of school that he/she will be disqualified if caught cheating.

5.2.3 Trust and power related to organization size and risk impact

From the examples of sections 5.2.1 (trust) and 5.2.2 (power) we can observe two dimensions that can be distinguished with both the personal and impersonal form of trust and power determining its predominant form: The number of actors involved (small, e.g. the parent-child situation or large, e.g. a school with many children) and the impact of risk (low or high). Fig. 5.4 shows these dimensions plotting the predominant trust relationship type into four quadrants depending on size and impact. This matrix will be used again later in this chapter to position the role of the SPG in arranging trust and power with some examples.

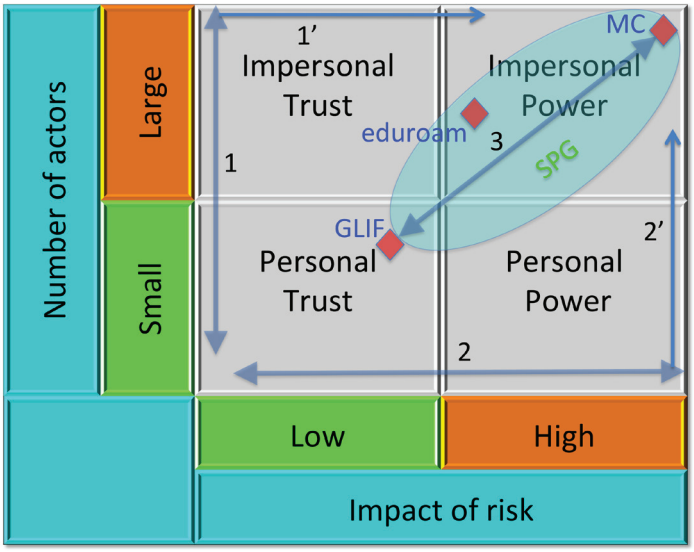


Fig 5.4. Predominant trust and power types related to number of actors involved and impact of risk

When moving your thought along line 1 from bottom to top, assuming consistent low impact, one can imagine that the amount of actors involved in a network of relationships determines the ease at which trust relationships can be maintained at personal level. When moving upward on line 1, one can see that an actor is limited in the ability to maintain personal relationships with other actors if the amount of relationships with different actors increases. An N^{th} actor has in theory to relate with N-1 actors to fully understand all existing tacit expectations of the group. Moreover tacit expectations are also likely to increase as more actors join. At some point all N actors are forced to make a part of the tacit expectations - that are understood as a group - explicit in the form of rules. This will allow the $N^{th}+1$ actor joining to understand the expectations in a more efficient way. This process is called institutionalization. At some point, the amount of rules institutionalized will outgrow the number of tacit expectations that can be maintained at personal level. Now impersonal trust becomes the predominant way to trust an actor's expected behaviour. Impersonal trust is used as a term because actors typically also include system trust next to institutional trust. For example, the group likes to identify new actors by asking for authority issued credentials such as a passport, licence, registration at chamber of commerce, etc. A relationship between a teacher of a small school and "ideal" pupils (that always can be trusted) can be placed at the bottom part of line 1. When the amount of ideal pupils increases, institutionalization of rules that need to be understood by ideal pupils becomes the predominant way of trust, moving the trust relationship type to the impersonal trust type quadrant. Note that with real pupils, the power of consequences will also be required to help determine the expected behaviour of pupils, moving the predominant trust type towards the impersonal power quadrant (arrow 1').

Line 2 is applicable to the parent child relationship mentioned earlier. For an ideal child - that always listens - the trust relationship can be placed at the far-left side of the line 2. If, however, a child's behaviour is impacting school results, the parent may decide to use its personal power by asserting consequences as the predominant way to trust the child. This moves this trust relationship type along line 2 to the personal power quadrant. Note that when results worsen, school (and in severe cases, the authorities) can become an active part of the relationship network of both parent and child. The position of such trust relationship will then to move to the impersonal power quadrant as the institutional power of school and system power of the authorities will become predominant. School and possibly authorities will take away personal trust and power from the parent (arrow 2').

When placed in this matrix, the trust relationship for example within the GLIF collaboration can be positioned in the personal trust quadrant. Participating organizations personally trust each other to act in the spirit of GLIF collaboration. Participants are expected to pool their excess resources for the collective good of their communities. A single page straw man charter constitutes the basis of understanding within the GLIF. This is because too much regulation within research communities is considered to be counter-productive. It is commonly understood that regulation and associated bureaucracy endangers the freedom and agility needed for innovations to happen.

The SPG can be seen as an attempt to institutionalize personal trust and power to an impersonal form that makes doing business more effective in managing impact of risk. It also allows collaborations to grow by providing impersonal trust using personal trust, created by initiatives such as the GLIF, as starting point. The SPG concept will help such efforts to move somewhere along line 3.

When considering the combination of a large number of actors and the large impact of risk (top right quadrant), regulation becomes increasingly important. In this sense Bachmann concluded that:

- *"in strongly regulated organizations, impersonal forms of trust and power tend to link into each other in such a way that powerful intra-organizational and environmental structures breed trust between individual actors in a highly efficient manner", whereas*
- *"in weakly regulated organizations, by contrast, individual efforts to establish cooperation between relevant actors in the organization become more important".*

The first regime fits the upper right quadrant whereas the latter regime fits the bottom left quadrant.

Bachmann noted that these organizational regimes form two extremes on a scale, where empirical cases can be found somewhere in between. Bachmann's observation intuitively fits the extremes of line 3 in fig. 5.4. Considering line 3 as a scale, we consider the SPG as a way to help facilitate the organizational transition from an informal and flexible initiative, mostly based on personal trust into a form that coordinates service delivery in a regulated way where power provides good reasons to provide trust. In this sense Bachmann confirms that: *"In strongly regulated organizations power primarily exists in the form of abstract rules and procedures. This form of power (that is impersonal power) is highly conducive to the production of institutional trust and system trust within the organization"*. In particular when organizations scale up, i.e. become more industrialized in managing economic value and involved risk, power defined by rules and regulation becomes predominant. By arranging rules and regulations, the SPG concept is intended to help organizations move to the upper right quadrant of fig. 5.4 as shown with the shaded area.

In order to recognize the contours of an SPG in the light of fig. 5.4, it makes sense to look at an extreme case, i.e. a large-scale, highly regulated case that has matured over many years. A case that succeeded in managing risk in such a way it grew into a trusted global organization. We believe that an example taken from the Payment Card Industry qualifies as such. We choose MasterCard (MC) as a qualifying example fitting the extreme case as shown in fig. 5.4. We could have chosen other examples such as Visa, Amex, etc. however as MC started as an association of a few collaborating banks, its evolution resembles the path of line 3 most closely. However, a SPG can also start the way Visa started as will be explained later. We will then use a networking example (eduroam) to recognize if the SPG contours fit.

5.3 Examples put in trust and power context

5.3.1 Payment Card Industry example

"Master Charge" (now known as MasterCard) started as an understanding between 14 banks forming the Interbank Card Association (ICA) in 1966. Unlike other payment cards introduced earlier (Visa, Diners, etc.) a single entity did not dominate ICA and its members. Member committees were established instead. These committees established rules for authorization, clearing and settlement, gradually creating more regulation introducing more impersonal power that allowed ICA to admit more members and manage the impact of risk.

MC arranges card payment authorization transactions to happen between four parties as shown in fig 5.5.

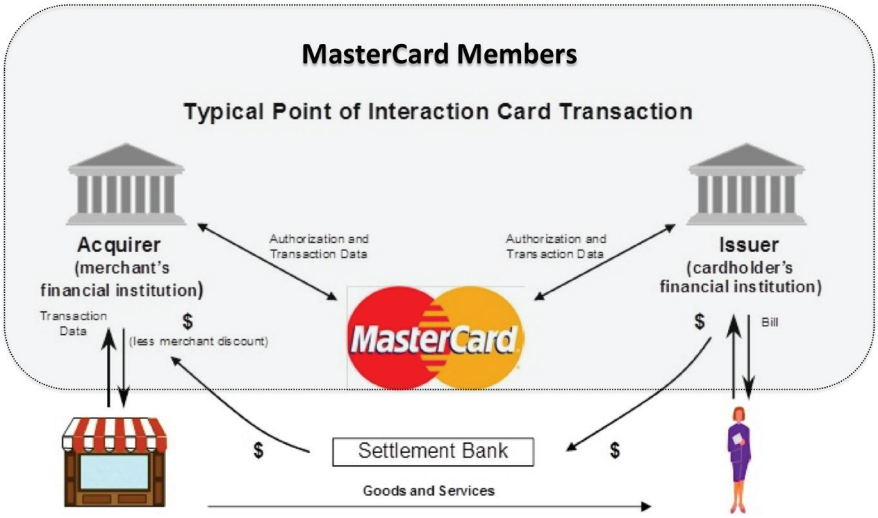


Fig 5.5. The role of MasterCard.

Cardholder (A) is issued a Card based on an agreement with Issuer (D). Amongst other things, the agreement determines the height of the payment limit that is extended to the Cardholder. Merchant (B) has an agreement with Acquirer (C) that arranges the authorization of a Merchant to accept a MC branded card and a Point of Sale (PoS) terminal capable of handling MC payment transactions. Note that Acquirers and Issuers are MC members. Merchants and Cardholders are users registered with MC members. In general, after the cardholder has presented his/her card, the Merchant's PoS terminal will send an authorization request to its Acquirer. The Acquirer will use the MC Interchange Network to route the request to the Issuer based on the card number. The Issuer is then requested to authorize the requested amount and returns an authorization code if the payment limit is not exceeded. Once the Merchant PoS receives a positive authorization, the Merchant will hand over the purchased goods and the

transaction is completed. The Merchant will receive the authorized amount (less a discount) into his bank account within a few days after the clearing & settlement process has taken place. To make this payment authorization system work between all involved parties, MasterCard Corporation Inc. has set up an elaborate set of rules for this Payment "Service Provider Group" consisting of competing but in this case collaborating financial institutes. Now that we better understand what trust means and know how an SPG is intended to help establish this, let us examine what it means that "Trust is necessary to allow each entity to "know" that the policy it is authorizing is correct" by considering Trust Notions 1 and 2 of section 5.1.1 as repeated in below table:

Trust Notion	Description
1	Users trust the predictability of the system's outcome as a whole
2	Collaborating entities trust each other to act in a correct, coordinated and predictable way

5.3.1.1 Considering Trust Notion 1

When accepting a credit card as a means to pay, the merchant (as a user of the payment system) must have *system trust* in the payment- & banking system as a whole that, after his PoS terminal receives an authorization for the requested amount, the amount will be added to his bank account before handing over the goods to the cardholder. It also means that the merchant must have the knowledge to trust that he has followed the correct rules and procedures *(institutional trust)* when accepting the card from the cardholder and obtaining authorization for the transaction. For example, acquirers in Europe increasingly expect merchants to use PoS terminals that can use the embedded chip in the credit card that allows a PIN number to be entered on the PoS by the customer rather than using the old system with the less secure magnetic strip and customer signature. Only when using PIN, the merchant knows he will not be liable to fraud in case the card was stolen. Such liability has been shifted to the merchant when the magnetic strip / signature method is used. When using the magstripe, an element of *personal trust* is still involved as it is then up to the merchant to personally trust the cardholder to decide to ask for a picture ID (relying on *system trust* provided by national authorities) to verify the signature. Lastly the cardholder has *system trust* in both the consumer protection laws and the payment card system such that he/she will not be liable for any fraudulent transaction that might appear on the card account.

5.3.1.2 Considering Trust Notion 2

By following the correct rules, regulations, bylaws, standards, etc. during its interactions with other financial institutes to correctly handle a transaction, each institute within the payment card system has the *institutional trust* that it will avoid damages or liabilities. For example, the

issuer should check and decrease the payment limit when authorizing the amount. If an institute chooses not to do so (e.g. by allowing default authorizations for amounts below a defined limit), all institutes know that this institute remains liable for the amount and cannot refuse the payment of the authorized amount during the clearing & settlement phase later in the process. Default authorizations will not cause damage to other banks. Also, all institutes must have enough liquidity to always be able to settle payments with other institutes. Every institute has the *system trust* that all other institutes are able to comply with this policy as all institutes reside under a National Bank that oversees compliancy.

5.3.1.3 Differences between Trust Notions 1 and 2

In the light of these observations, we can see that RFC2904 only targets trust that is formalized in a Service Agreement (see fig. 5.2). This trust is embedded in the impersonal trust and power (rules) provided by a body such as MC. Note that Trust Notion 1 targets trust between a Person and an Organization, whereas Trust Notion 2 targets intra-organizational trust. Clearly Trust Notion 1 involves both personal and impersonal trust elements, i.e. the organizations reputation next to impersonal trust that is arranged in the agreement with the user. Both are fundamental elements why users trust payment cards. Although difficult to separate [NOO1], the willingness to rely on reputation tends to involve emotional factors, whereas relying on agreements tend to involve rational factors. MC must therefore use its institutional power to manage both Reputation and Agreements with users to breed trust towards its users as shown in fig 5.6.

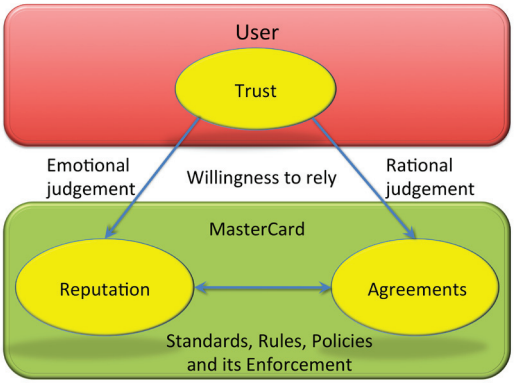


Fig. 5.6 the elements why users trust MasterCard.

Trust Notion 2 involves mostly impersonal trust. This as agreements between MC and their members arrange that members do not have to personally trust each other anymore as individual members. Sufficient regulation conduces the necessary impersonal trust as noted by Bachmann in section 5.2.3 regarding abstract rules and procedures in strongly regulated organizations. Agreements embedded in the institutional trust created by the MC Rules allows members to

trust each other as shown in fig 5.7. All members trust MC to ensure new members can be trusted after joining. This trust is the basis for all interactions with the new member.

MC is responsible for managing the reputation of the member group as a whole and must have the power to be able to do so. Much of its regulation has been targeted towards avoiding reputation damage. Once a member signs its membership agreement, declaring it will comply with all MC Rules, MC Operating Regulations subsequently ensure up-to-date knowledge such that member banks can maintain compliancy.

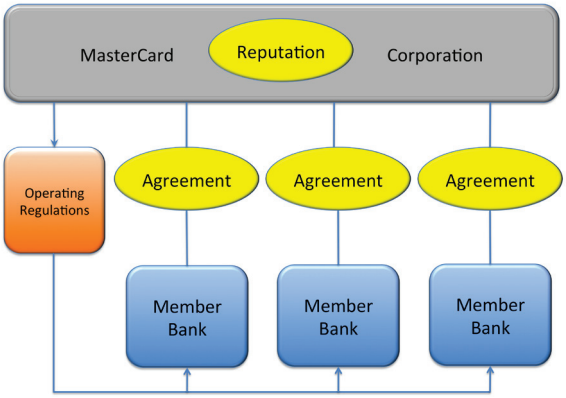


Fig 5.7: Why members trust each other via MasterCard.

5.3.2 Eduroam: A network related example of a SPG.

Eduroam [EDUR] allows students, researchers and staff from participating institutions to obtain Internet connectivity across campus and when visiting other participating institutions by simply opening their laptop. Eduroam allows participating research & education institutions, known as eduroam Service Providers (SP's), to provide access to students from any other participating institute that acts as Identity Provider (IdP). Eduroam is a federated roaming service that provides such secure network access by authenticating a user with their own credentials issued by their IdP. A group of National Research & Education Networks (NRENs) are in essence providing this service for their participating educational institutes under the eduroam "brand" arranged by TERENA. Fig. 5.2 is in essence the model used in eduroam, where a User Home Organisation is the IdP with an agreement with a User (student, researcher, etc.). SP's and IdP's have agreements with each other being a participant (group member) of eduroam.

Eduroam work started in 2003 as the TERENA Task-Force Mobility [TFMO] with the overall aim: "to assist in fostering trust between academic institutions and between NRENs so that these critical relationships can encourage active participation, and the development of roaming services". In their first policy document [TEPO], the taskforce clearly recognized that *"To facilitate the interest shown*

in roaming services it is important that policies are put in place at appropriate levels to ensure that benefits remain whilst threats and risks are minimized and managed within acceptable levels." In this document, task force members established a set of rules that outlined agreements to be signed between TERENA (facilitating in this case eduroam as SPG principal) and each participating NREN (Intra-NREN roaming policy). It also outlined agreements between an NREN and their participating national institutes (NREN level policy). In these institutionalized set of rules, NRENs and participating institutes were made responsible for administration, monitoring and enforcement of a number of rules. For example, NRENs were made responsible to write guidelines for participating institutions to assist them in draughting local site and user policies to ensure compliance with the roaming service agreements with their NREN. Participating institutes must report any security issues or fraudulent activities and log authentication sessions and network access session and be able to trace a user for both security and capacity planning purposes. Over the years, the initial agreement and their rules has evolved into a compliance statement [EDUC] organizing a confederation (federation of federations) that is funded by Géant [GEAS]. A Global eduroam Governance Committee has been made responsible for the rules contained in the compliance statement and is also responsible for the final ruling on disputes that cannot be resolved within the community. Eduroam is available now in 60 territories worldwide.

5.3.3 Summary

Within eduroam, the first policy document was aimed at fostering trust between participants. It institutionalized *impersonal rules* and provided TERENA the *impersonal power* to admit by having NRENs sign agreements to participate in providing eduroam services. NRENs on its turn were made responsible to administer and enforce impersonal eduroam SGP rules towards its national institutes such that all participants had the correct knowledge to interoperate with other eduroam participants and understand what to do to minimize threats. Eduroam has a mechanism to allow their rules to evolve in a coordinated fashion. Fig. 5.4 shows the position of eduroam as SGP. Compared to MC, it manages less risk but has a significant amount of worldwide users; although less than MC. Note that fig. 5.4 does not attempt to provide an absolute scale.

Trust within MC is also conduced by a set of rules originally established by ICA. These rules have evolved over many years by MC continuously evaluating and enforcing them. Members give MC the power to judge and sanction non-compliant behavior. This *impersonal power* ensures correct execution of payment authorization requests by making sure each of the entities precisely knows about the correctness of the policies it executes and also understand the consequences of non-compliance. This, however, does not always guarantee that *personal trust* (reputation) of its users can be maintained in individual cases as this also depends on personal expectations, new ways of fraud, changes in consumer laws, etc. MC has therefore the means and power to manage re-occurrences and changes by allowing it to continuously update its rules and regulation. With the establishment of ICA in 1966 MC started as an SPG on the left side of arrow 3 of fig 5.4 and slowly evolved to the right of arrow 3. Amongst a growing number of members managing more and more risk, MC maintained the necessary trust needed to allow each entity

to "know" that the policy it is authorizing is correct. It was only able to do this by introducing more and more impersonal power based on its rules.

5.4 Conceptualizing the SPG Framework

In this section we will conceptualize the SPG Framework from the observations made in previous sections. Here we will both consider the MC example and described AAA Authorization Framework and note that similar observations can be made when considering the eduroam example to verify the thought around our framework. The MC Rules [MCRU] are in essence rules that describe what it means to be a "Member" (or more recently a "Customer") of MC. As these Rules change from time to time, we examined the MC Rules as they stood per July 2011. Please note that the MC rule numbering, used in the next sections, refer to rules from the document as it stood then. The eduroam compliance statement [EDUC] in essence describes what it means to be a participant in the eduroam confederation.

5.4.1 Additional Terminology

In addition to the terms described in section 5.2, there are a few additional terms (table 5.2) for MC and eduroam that need to be defined and are used in following sections.

Context	Term	Description
MC	**Member, Membership**	A financial institution or other entity that has been granted membership in and has become a member of the Corporation in accordance with the Standards. "Membership" means membership in the Corporation.
MC	**Standards**	The Amended and Restated Certificate of Incorporation, Bylaws, Rules, and policies, and the operating regulations and procedures of the Corporation, including but not limited to any manuals, guides or bulletins, as may be amended from time to time.
MC	**Control**	As used herein, Control has such meaning as the Corporation deems appropriate in its sole discretion given the context of the usage of the term and all facts and circumstances the Corporation deems appropriate to consider. As a general guideline, Control often means to have, alone or together with another entity or entities, direct, indirect, legal, or beneficial possession (by contract or otherwise) of the power to direct the management and policies of another entity.

Context	Term	Description
eduroam	**Identity Provider** (**IdP**)	An entity that is responsible for user credentials and operation of an authentication server for eduroam access for these users. IdPs are in some regions also known as "Home Institutions"
eduroam	**Service Provider** (**SP**)	An entity that operates an access network on which eduroam users are admitted to access Internet services once they are successfully authenticated by their IdP. SPs are in some regions also known as "Visited Institutions".
eduroam	**Roaming Operator** (**RO**)	The entity that operates the eduroam service for a country or economy and that is recognised as such by the RC to which it belongs or, in case the country or economy is part of a geographic region for which no RC is established, by the GeGC. The RO may be a National Research and Education Network operator, for example. ROs are sometimes referred to as the "eduroam operators".
eduroam	**RC**	An entity that consists of a cohesive set of ROs serving a geographical region and that is recognised as such by the GeGC. The "European eduroam Confederation" is one example.
eduroam	**GeGC**	The TERENA co-ordinated Global eduroam Governance committee (GeGC), comprises of representatives from ROs and RCs; they have written the compliance statement.

Table 5.2: Additional MasterCard & Eduroam terms

5.4.2 Analyses of MasterCard rule examples

By using some examples taken from the MC Rules, let us consider of how powerful MC is in interacting with its members and how various types of trust and power play a role.

MasterCard Corporation was (until recently) a membership organization for financial institutions. When applying for membership, MC has the power to determine if an organization does fulfil all its requirements. Members must be financial institutes that are recognized by a National Authority. Here, MC has *system trust* in competent National Authorities.

Issuers must have a Licence from the Corporation before they can issue cards to Cardholders. MC has the *institutional power* to arrange - via Licences - the area's in which issuing activities may take place.

The power of MC goes beyond its members: An Acquirer must have a Merchant Agreement with their Merchants before the Merchant is authorized to accept Cards. According to the rules, MC has the *institutional power* to determine what provisions members must put into their agreements with merchants or cardholders and as such be in control.

From the MasterCard Rules we can observe some additional rules showing the power of the organization: These examples show how MC manages Trust Notion 2 (see section 5.1.1.) in an attempt to earn Trust Notion 1.

MC and the Acquirer have the power to enforce the correct use of their Brand Marks as it both may audit the Merchant's activities when it uses the MC Brand Marks pursuant to a Merchant Agreement. An Acquirer is in violation of MC Rule 5.1.1: *Before entering into, extending, or renewing a Merchant Agreement, an Acquirer must verify that the Merchant from which it intends to acquire Transactions is a bona fide business* - if it knows a Merchant sells illegal goods.

With Rule 5.3: *Each Acquirer must monitor on an ongoing basis the Activity and use of the Marks of each of its Merchants for the purpose of deterring fraudulent and other wrongful activity and to ensure ongoing compliance with the Standards* MC has the power to ask the Acquirer to monitor its Merchants. Rule 5.10 shows how far reaching this power of MC can go: *If the Corporation becomes aware of a Merchant's noncompliance with any Standard, the Corporation may notify the Acquirer of such noncompliance and may assess the Acquirer, and the Acquirer must promptly cause the Merchant to discontinue the noncompliant practice.* Clearly the Acquirer itself must have the power to monitor and terminate the Merchant agreement if the Merchant does not discontinue the noncompliant practice.

Note that the MC Rules defines its Standards that not only include the Rules itself but also its bylaws, operating regulations, policies, etc. Standards are based on the Rules and contain more details as a result of the interpretation of the Rules.
It is important to note that MC considers a failure to comply with any Standard, to adversely affect the Corporation and its Members. It also undermines the integrity of the MasterCard system. The Integrity of the System is an important factor in the willingness of all involved parties to rely on it, i.e. trusting (both personally and impersonally) the reputation of the system.

5.4.3 Combining impersonal power and authorization framework into the SPG concept

In this section we will use the previous observations to conceptualize a framework describing how the SPG contributes towards building trust from impersonal power mostly formed by its institutionalized Standards.

5.4.3.1 Organizing the Institutional Power using the Trias Politica

Observing the MC Standards, i.e. the institutional power of MC Corporation, made us wonder how such power, and the ways it comes to play within MC and its member organizations can be organized in a more abstract way. A well-known abstraction, the "Trias Politica" by Charles de Montesquieu [MSQU] recognizes three different types of power: Legislative, Judicial and Executive power. By classifying the MC Rules into these three categories we were able to observe what parts

of power are given to MC members and what power resides in the domain of MC Corporation. We also could see what type of rules MC established in each of the power categories. We used below interpretation to classify the MC Standards. We could then find examples of MC Rules fitting into each category. A subsequent study of the eduroam rules provided more confidence in the applicability of this approach.

Power		
Legislative	**Judicial**	**Executive**
Power to make rules.	Power to determine interpretation of rules.	Power to administer and enforce rules.

Table 5.3: Used interpretation of Trias Politica

Note: with "power to administer" we refer to the ability to translate the Standards into information needs, policies, procedures such that its result can subsequently be applied to- and operationalized by (automated) processes with authority. We will use the term "rulemaking" instead of "legislative" as legislative implies creation of law by a deliberative assembly, whereas organizations, such as MC and eduroam create rules rather than law.

5.4.3.2 Functional Level perspective

The RFC2904 AAA Authorization Framework describes a functional level that handles policy-based decisions authorizing access to resources. The policies are based on common rules and whatever has been agreed between the parties involved in the decision. As such, we can recognize the authorization transaction handling functions as the 'Policy level' that sits in between a level that determines what these policies are required to be (Business Level) and a level that represents the resources and its controls what these requests and applied policies are about (Operational Level). Table 5.4 contains a high level description of the main functions of each level.

Level	Description
Business Level	Builds and maintains a business structure that delivers defined services according to established rules and agreements between providers acting as a group. Responsible for administering and enforcing rules. Accountable for service delivery towards users.
Policy Level	Responsible for handling service authorization transactions by executing administered policies and controlling the operational level. Provides information that allows monitoring and enforcement.
Operational Level	Responsible for delivery of authorized service according to a service request and provides the proof of correct delivery. Providing information that allows monitoring and enforcement.

Table 5.4: Functional Levels

The Business, Policy and Operational levels have agents within each member's organization that are capable of putting the policies into operation driven by automated protocol exchanges. This will be further explained after we have explained more about the SPG framework.

5.5 The SPG Framework

As will be motivated by referring to actual rules [MCRU], a further study of the MC Standards and previous considerations lead us to the compilation of the SPG framework as illustrated by fig's 5.8, 5.9 and 5.10. MC and its Members are considered to be a form of SPG. It combines the two perspectives describe in section 5.4 with an organizational perspective that describes establishing the SPG. Considering the eduroam rules [EDUC] provided evidence that the framework also fits in this context.

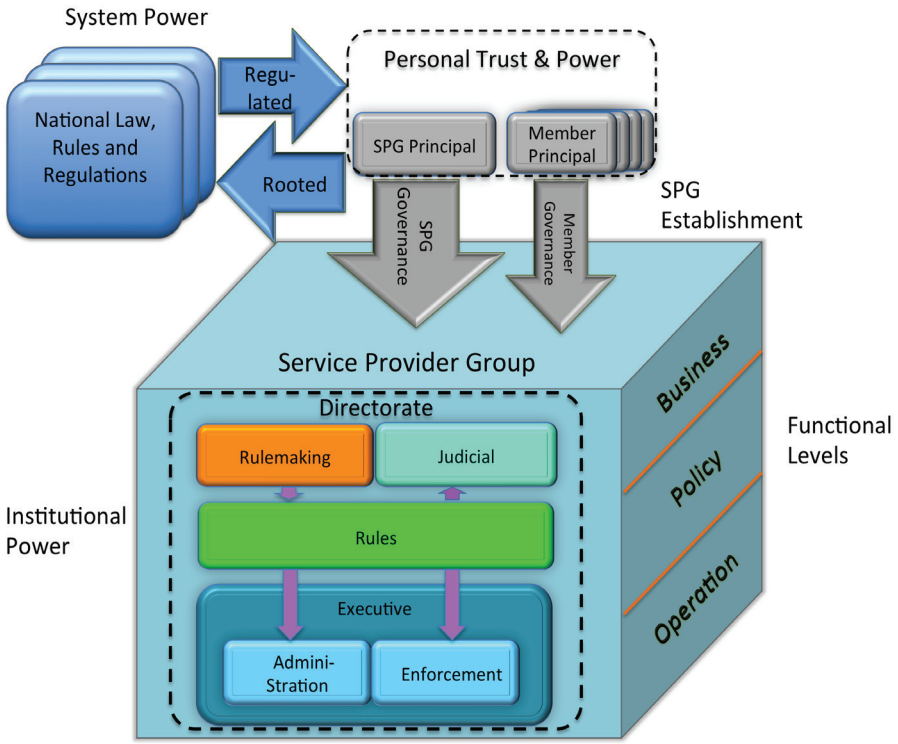


Fig 5.8. The SPG Framework high-level perspective.

5.5.1 High-Level Perspective

Fig. 5.8 shows the High Level SPG concept, considered from three perspectives: The establishment perspective and the earlier described power and functional perspectives. We will now consider the establishment perspective.

A Principal is the actor of a service provider that can be held accountable for all its business activities and decisions. The assumption is that Principals, personally trusting each other, will establish a SPG after recognizing and agreeing on a mutual business benefit. MC was established as such by 14 banks as the International Card Association in 1966. The SPG could also be a single entrepreneur using his network of trusted business partners. Bank of America established BankAmericard (later to become Visa) this way by starting with licencing their card service to other banks in 1965 [VISA]. Eduroam was based on the recognition within TERENA that there was a need for allowing guest students hassle free network access using their own IdP credentials to authenticate from.

Based on consensus for a common business strategy, Principals establish the SPG. During the formation of a SPG, Principals will either elect a SPG Principal (typically based on a combination of personal trust and power i.e. size of the contribution to the group) or the SPG Principal is the founding entrepreneur. The SPG Principal is subsequently held accountable for the activities, agreements and service delivery of the SPG as a whole. A Member Principal will determine what part of its resources will be made available to the SPG. The SPG Principal must define requirements for such contributions.

When creating its organization, a Principal establishes a Directorate role as an efficient way to coordinate its activities. A Principal (via its Directorate) holds the institutional power of its organization. Such power is used to control the three functional levels of an SPG organization. The SPG consist of multiple SPG member organizations and a single SPG governing organization.

When considering MC as a SPG, the chairperson and board of directors of MasterCard Inc. can be considered the Principal of the SPG governing organization. The governing organization's Principal appoints a CEO as its Directorate head and is made responsible for establishing the three institutional powers (Rulemaking, Judicial and Executive) that will govern the SPG as a whole. This as the rulemaking power governs the behaviour of MC and its members by creating the MC Rules. The MC Executive Power is based on these Rules.

When considering Eduroam as SPG, TERENA acted as Principal that created a taskforce that performed rulemaking. Much of the executive power was given to the NRENs to oversee the national institutions. The later establishment of the Global eduroam Governance Committee formally can be seen as the SPG Directorate that established the Rulemaking and Judicial element. The Executive elements have been established as a confederation organizing the RC's / RO's (see table 5.2) as members that sign an agreement with the SPG Principal (Géant/Terena).

Any organization's Principal must comply with National Law and their legal requirements. Here the SPG governing organization's Principle makes its Directorate responsible to oversee that it and its members activities always comply with applicable laws, even if a Member's National law and regulation contradicts SPG Standards. MC Rule 3.2 states to this respect: *Each Member at all times must conduct Activity in compliance with the Standards and with all applicable laws and regulations.* MC and Member organization activities are as such rooted in applicable system power via the responsibility of the Principal.

Also National Authorities change regulation from time to time. For example, nowadays MC members may be required by their national authority to conform to BASEL III guidelines [BAS3]. The SPG and its members are then also regulated by system power of the national authorities where applicable. Fig. 5.8 shows therefore two arrows between the Principles and the national Law, Rules and Regulations.

5.5.2 Organization Viewpoint

Fig. 5.9 provides more detail describing the basic concepts shown in Fig. 5.8 considering the SPG Framework from an organizational viewpoint. It shows details that are applicable to a participating Member organization within the SPG. The Business level contains the Principal that holds the three powers via its Directorate. The Executive Power administers and enforces its activities using a Business Support Agent (BSA). The framework distinguishes Administration and Enforcement as separate elements of the Executive Power. Executive Power is defined as the authority to ensure activities are carried out according to the rules whilst facing potential consequences for non-compliance. Interpreting rules and implementing the resulting policies governing the operation and decisions within the relevant processes we define as administration. Keeping oversight over the outcome of processes and resulting services delivered we define as enforcement. Members Principals are put in control by the SPG governing organization for its activities that must be performed and overseen according to the community rules (standards). This can be observed from MC Rule 1.5.5-1 [MCRU]: *a member must at all times be entirely responsible for and Control all aspects of its Activities, and the establishment and enforcement of all management and operating policies applicable to its Activities, in accordance with the Standards.* The term Control (table 5.2) implies that a Member must have the power to do so (even if chosen to outsource parts of its activities).

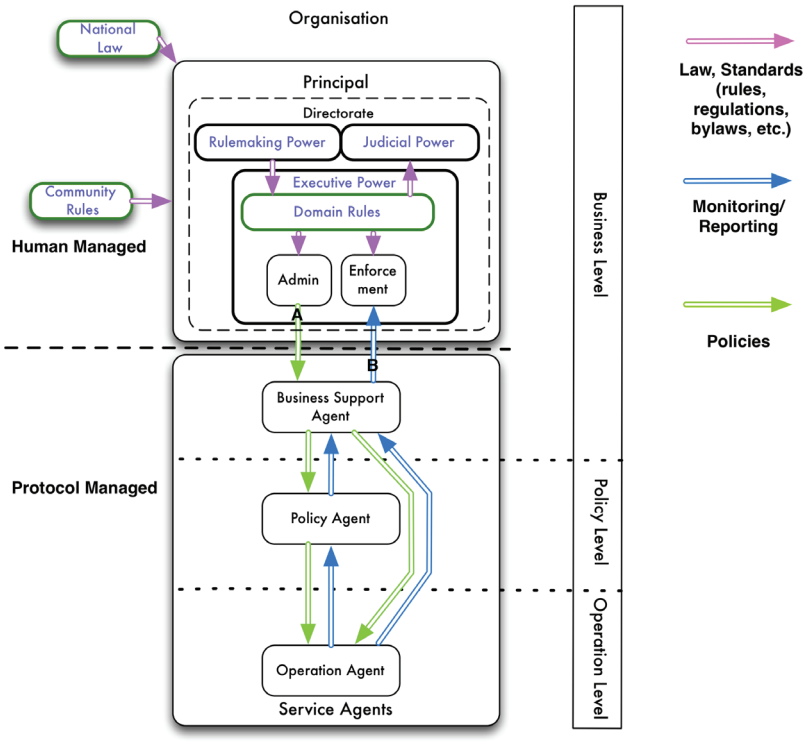


Fig 5.9. The SPG framework organization viewpoint

Within eduroam, the community rules clearly makes the Principal of a Roaming Operator (RO) responsible: Rule 4.1 of its compliance statement [EDUC] says: *The RO is responsible for ensuring the eduroam service operation within a particular country or economy. Rule 4.3 states: The RO has the authority to determine the eligibility of eduroam IdPs, being organizations engaged in research and/or education, in its country or economy.* Here the RO can make its own ruling to determine the eligibility of an IdP, however it must take national law into account to determine the legitimate status of an organization as being involved in research and/or education.

A BSA is the overall management entity allowing a human interface into the system providing control, monitoring and reporting functions regarding the services provided. Administering the delivery system with the necessary policies based on applicable rules performs the control of the service delivery. The policies administered are based on the rules autonomously determined by the domain itself (domain rules) considering the institutionalized community rules provided by the SPG governing organization and National Law. The service delivery system is build using functional service agents: The BSA, Policy agents and Operational Agents that handle service authorization transactions and service delivery. The Member organization administration controls the BSA by determining for example what part of the available services will be assigned to the SPG, what its usage limits are, what users can request as a SPG defined service, what kind of information needs to be reported and/or enforced, etc. This type of information is provided via green arrow A. Via blue arrow B, the BSA will also provide information that need to be enforced according to the rules. Within a service provider domain, a BSA can (automatically) configure different forms of Policy- and Operational agents. As Fig. 5.3 imagines, a BSA could for example configure various types of Network Service Agents (NSAs) [GFD173, KUDO] the OGF NSI working group is suggesting. The NSI NSA concepts then implements the protocol managed policy- and operational levels of a Network Provider Group (see section 5.1.2.2).

5.5.3 Organization Interaction viewpoint

Fig 5.10. shows how the SPG governing organization interacts with its Member organizations. Member organizations that provide services (e.g. in the MC model the Acquirer) or Member organizations that register and represent users (e.g. the Issuer).

The Administration side of the SPG Executive Power refines the SPG Rules by adding bylaws, operational regulations, policies, etc. forming the SPG community Rules (standards). By promulgating and enforcing these community rules (that change from time to time) the SPG governing organization as such ensures that Members "`know when the policy it is authorizing is correct.`"

Hereto the Principal of a Member has signed an agreement (arrow 1) with the Principal of the SPG. Both MC rules (see table 5.2 definition of Standards) and Eduroam rules have provisions that allow rules to be amended. Eduroam states in this respect: "This document is subject to change by the Global eduroam Governance committee (GeGC), based on feedback from ROs, RCs or individual eduroam users."

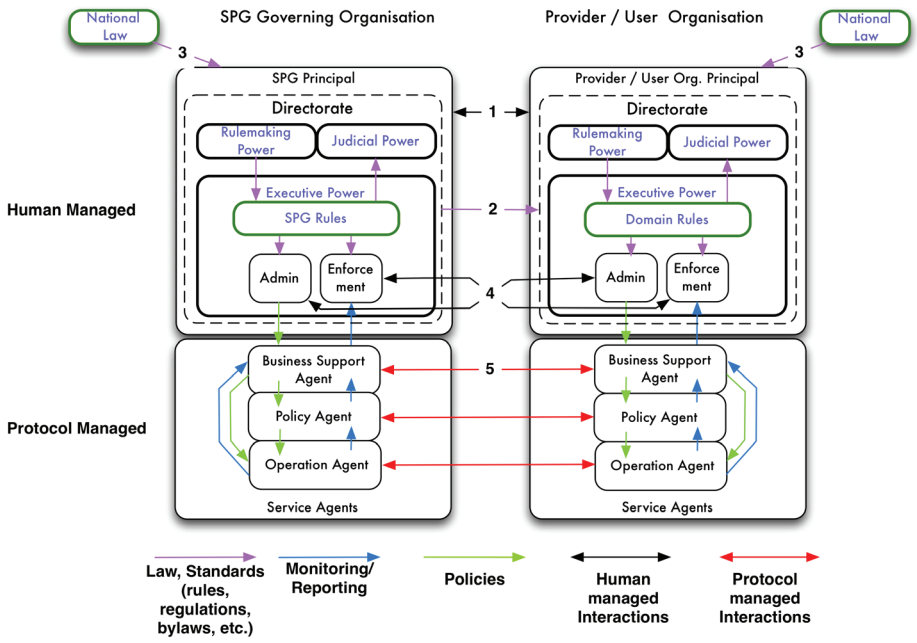


Fig 5.10. Interactions between SPG and its Members (Provider/User Organizations).

This agreement is the root that builds trust between organizations as was illustrated in fig. 5.7 of the MC example. As result, the SPG Directorate provides the SPG community rules to the SPG Member Directorate (arrow 2) such that it can be integrated into the policies administered to the member's BSA and allow its enforcement. Note that the SPG Executive Power can also influence a SPG Member's Users behaviour as can be seen from MC Rule 5.1.1 and 5.3 where Merchants (as the User of an Acquirer) are not allowed to sell illegal goods and accept a credit card as payment method. The eduroam compliance statement for ROs contain statements that govern their SPs and IdPs (as users of the RO) by stating for example: *"By signing this document, an RO commits to ensure that the eduroam IdPs and eduroam SPs in its country or economy implement and adhere to the rules set forth herein."* Here the RO is expected to have executive power (administration & enforcement) over its IdP's and SPs.

MC Rule 1.5.5-3 states that a Member must ensure that all policies applicable to its Activities conform to the Standards and comply with applicable laws and regulations. This Rule states that the Members Principals must take into account applicable national laws and regulations and must also allow their activities to be regulated as shown by arrows 3. Member administrations must interpret their own Rules and the SPG Standards in the context of their National law and regulations and translate and implement them accordingly into their own policies. For example an Acquirer should create a policy for accepting credit card transactions that are used for commercial gambling. Accepting such transactions is depended on national or federal law (such as the US Unlawful Internet Gambling Enforcement Act [FRS]). Acquirers are responsible to implement a policy accordingly not to accept transactions from Merchants that fall under such

law. When the outcome of these policies is sufficiently enforced, it ensures the implementation of both system trust (national laws and regulations) and institutional trust (i.e. MC Standards) within the SPG avoiding liabilities. As seen earlier (rule 4.3), eduroam requires IdPs to be research and education institutes. There may be legal requirements to be recognized as an education institute.

The Enforcement side of the Executive Power is responsible for monitoring compliancy with the Rules and its refinements and should signal any issue found based on information that the Administration requests and Enforcement receives. Both Member- and SPG Directorates must oversee such requesting, signaling and enforcement as shown in fig. 5.10 with arrows 4. Business Support Agents may support such process, using automated protocols exchanges that automatically request and/or report information (arrow 5). The BSA will manage where such policies must be applied in the Policy- and Operation Agents.

The SPG Directorate Judicial power handles any disputes regarding the interpretation of rules and standards for SPG defined services and will decide on possible (disciplinary) measures. MC Rule 3.1 states to this respect: *From time to time, the Corporation promulgates Standards governing the conduct of Members and Activities. The Corporation has the sole right in its sole discretion to interpret and enforce the Standards. The Corporation has the right, but not the obligation, to resolve any dispute between or among Members including, but not limited to, any dispute involving the Corporation, the Standards, or the Members' respective Activities, and any such resolution by the Corporation is final and not subject to appeal or other reviews.*
The eduroam compliance statement says in this respect: "In case of a dispute regarding the status of an entity (IdP, SP, RO) in the eduroam service that cannot be resolved by the responsible RO or RC, the GeGC will give the final ruling"

Other SPG governing organization activities could include soliciting, sales and marketing, admitting and administering members, defining services, keeping oversight and enforcement, handling complaints, etc.

5.5.4 Business Service Agent Responsibilities

A Principal will, by using its Executive power, delegate responsibilities (green arrows) to the BSA. The BSA will make one or more Policy and Operational Agents responsible for handling Service Authorizations and correct Service Delivery. The BSA is responsible for the Service Delivery architecture, i.e. the layout and management of Service Agent functions and their relationships across its infrastructure. Policy Agents are responsible for authorizing Service Requests based on administered policies and enabling Service Access. The exact meaning (semantics) of a Service Request (object) can be largely determined by the administered policies. The conditions of Service Delivery that are handled by Operation Agents are defined by the result of the administered policies. BSA's are expected to have the all required knowledge about how the administered SPG rules and policies can be implemented and enforced within its organization. As such a BSA is responsible for implementing and enforcing the correct semantics of service request objects.

The Business Support Agent is responsible for collecting, aggregating, processing and reporting information to the Enforcement part (blue arrows). Such information originates from the Policy and Operation Agents that deliver Services. The Business Support Agent manages such collection and translates it into appropriate signals that trigger enforcement. It determines where Policy and Operational agents are required to collect what information and subsequently configures these agents with the correct policies.

When policies are correctly administered and enforced policies are *"known to be correct"* to handle the responsibility of authorizing a service request and putting corresponding services into operation. This "knowledge of being correct" is an important observation that influences the information need required in the protocol interactions exchanging request- and control objects between Agents.

5.5.5 Importance of SPG Standards for the information need within protocol object exchanges

The more the Service semantics are handled by the impersonal power of standards via correctly implemented policies, the less need there is to communicate specifics about a service by protocol objects during the creation and fulfilment of a request. For example: The semantics and attributes describing a "Gold Service" may be entirely defined by the standards and its implemented policies. Communicating the fact that "A Gold Service" is needed between A and B for a specific time window may be sufficient to fulfil such a request. All SPG Members understand via the impersonal power of the standards precisely what "Gold Service" means and exactly know how such service should be provisioned using its policies. After making a request to the policy agent of an SPG member or the SPG itself, a user may be handed back a signed service reference by means of an abstract token that the Service has been arranged as requested as described by Gommans [GOM8]. What such token means and how it should be created and subsequently treated could be entirely defined by SPG standards. Once inserted in the service infrastructure it may mean "give me Gold Service for the next 10 minutes". Each member is able to recognize such at token when enforcing individual service accesses across multiple SPG members. Each member may interpret a token differently as long as the result matches SPG requirements. A Bronze token implying "good for best effort service" may receive high available service in one domain whereas it may receive non-redundant services in others. This can be done because policies *are known to be correct* and therefore this knowledge does not have to be repeated within protocol information objects. This allows token mechanisms to be an efficient way to communicate authorizations across multiple domains once SPG standards are in place. However, as motivated in RFC2904, there are many other sequences available to request and deliver an authorized service. More work is needed to value each possible sequence given the fact that policies are known to be correct.

5.5.6 Reputation Management

Agents also must provide the necessary accountability information (blue arrows) such that the enforcement organization part of the executive power can keep oversight and enforce SPG standards. This is an important aspect allowing the SPG governing organization to use credible impersonal power amongst SPG members. It also provides the ability of the SPG governing organization to manage its reputation towards the SPG users and amongst SPG members. SPG members may be asked to report to the SPG governing organization and/or allow assessments to be performed. See section 5.4.2 for MC examples. The SPG governing organization must be allowed to perform such assessments. What information could be asked to report is defined by SPG Standards. SPG Members are free in arranging such information themselves, as long as it fulfils SPG requirements. Eduroam compliance statement 4.9 states in this respect: *"The RO MUST make sure that the eduroam IdPs and eduroam SPs in its country or economy maintain sufficient logging information to allow the user identification process to terminate successfully".*

5.5.7 The SPG framework applied to connection oriented networking

Let us consider an application of the SPG framework in the case were a group of network providers are collaborating to provide connections as shown in NSI example (fig. 5.3) of the introduction. Let us assume that the principals of participating provider organizations, owning suitable lambda's and/or exchange points, have found business reasons to create a Network Provider Group (NPG). As said in the introduction, the NPG is an incarnation of the SPG framework applied to connection oriented network provisioning. The principals decide to create a NPG governing organization by appointing a NPG principal. The NPG principal establishes a NPG governing organization that is made responsible for coordinating the NPG activities using a directorate. The NPG directorate establishes institutional power by defining standards (rules, operating regulations, bylaws, etc.) for its members and defines under these standards what a service is and what delivery of a service means. Members have to agree to sign a service agreement with the NPG governing organization and declare it will comply with the NPG standards that will change from time to time. Users, registered with each member are potential users of the NPG provided services. The user relationships will remain the responsibility of an NPG member acting as user organization. Users, via its user organization, can now request NPG services. NPG policies and terms become embedded in the service agreement with the user such that the user has an understood contract with the NPG governing organization represented by the NPG member. The member now acts as an NPG agent for services provided by the NPG. The NPG principal is ultimately accountable for services delivered by the NPG. Members are accountable to the NPG governing organization for the quality of delivered service contributions. Agents (e.g. OGF NSI-WG defined NSA agents) are configured and used to handle the responsibilities and accountabilities that provide the group services using policies.

5.6 Future Work

We have shown a high level framework for a SPG that certainly needs more detailing. Future work is needed to describe in more detail what each power typically comprises of by studying more cases like MC and eduroam such as eduGAIN, EGI and various Géant connectivity services. Also the functional levels need more detailing in terms of functionalities and interworking with for example entities as described in the Network Service Architecture work of the OGF NSI working group and work that is done by the authors on Network Provider Groups.

The fact that SPG rules are administered as policies to Service Agents that *are known to be correct* is an important concept that needs further investigation. It plays an important role when determining the information and security needs for protocol objects being exchanged using sequences such as described by RFC2904.

The SPG framework is expected to be generic enough to be applied to many collaborating Service Provider cases as can be found within the infrastructure cloud arena build on converged infrastructures. Future studies are performed into these cases.

5.7 Conclusion

Increasingly, organizations rely on multi-domain converged infrastructure services performing their research and development or doing business. These services are composed of an aggregation of (competitive) individual infrastructure services. Such service composition is similar to competing banks offering Card Payment Services or providing worldwide eduroam WiFi Internet Access services. Such services can only be provided by a collaborating group of autonomous service providers. The delivery of such end-to-end services needs coordination and oversight to ensure quality, manage risk and liability. The willingness to rely on such services is associated with trust. Trust in the chain of services becomes a chain of trust. When any part in such a chain fails, the trust in the service as a whole fails.

Power, in the form of impersonal rules, is an efficient way to conduce trust amongst large number of members of a Service Provider Group. In strongly regulated organizations, power primarily exists in the form of abstract rules and procedures. This form of power (that is impersonal power) is highly conducive to the production of institutional trust and system trust within organizations. As trust is not absolute, power is needed and must imply realistic consequences for non-compliance.

With MasterCard as an example of a SPG we can see that its rules provide knowledge to all its members and users such that everybody understands how the system should be used correctly. Intentional misuse (fraud) by users and/or service providers is detected and handled in a powerful way, but also in a way that users, when obeying the MC rules and regulations, are not harmed as MC has judicial power. When Cardholders, Merchants, Issuers and Acquirers

know that the correct policies have been used during the authorization, everybody can trust that the end-to-end service will work reliable: Cardholders will only pay for ordered and received products, Merchants always get paid for the delivered goods or services and financial institutes have an agreed and viable way to manage and minimize fraud. The power of MasterCard will take the weak parts (non-compliant parties) out of the chain by excluding them as Cardholder, Merchants or financial service provider.

An organization structure with role for a governing organization was envisaged by Ian Foster for Grid infrastructures and by Kees Neggers for Network infrastructures. Helped by statements from NIST and Gartner and directions open Cloud standards are heading, we were encouraged to investigate how such a role can be described as a framework for a multi-domain service provider environment.

From observing MC we derived a framework for a Service Provider Group as a way to efficiently provide impersonal power needed to conduce trust amongst service providers such that all involved entities "know" that the policy it is authorizing is correct. The SPG Framework targets the business issue of organizing such trust between service providers that can only deliver a service to a user if they collaborate. The roles of creating, executing (administration and enforcement) and judging SPG Rules are essential organizational entities. They must be established to provide and maintain rules that are accepted to give a SPG governing organization the necessary power and credibility.

The Service Provider Group framework recognizes that it must be set up by a SPG Principal that obtains its mandate from Member Principals. In this phase personal trust between founding members is most important. When technology allows automated policy based setup and/or increasingly more participants join the collaboration, the SPG Framework is a way to arrange *impersonal power* that takes away the need to *personally trust* people to coordinate group activities. We have argued that a Service Provider Group is a concept that arranges *impersonal power* by establishing rules that can be translated into policies by administrations and applied to Service Agents of participants by using a Business Service Agent as linking element. Business Service Agents ensure with Policy and Delivery agents that executed policies can be trusted as *"known to be correct"*.
Assuming that the knowledge about policies is correct, has important implications for the information needed inside protocol objects that are exchanged to authorize and deliver services. The more knowledge the administered policies provide, the less need there is to communicate such knowledge inside protocol objects. This allows the exchange of abstract or simple tokens between service providers as proof of service authorization.

MasterCard, that initially stood example for the SPG framework, has proven to be a way of providing and maintaining trusted end-to-end services for the benefit of both customers and service providers. The SPG governing organization should keep both benefits in mind to be viable. We motivated that the SPG framework is also applicable to multi-domain network connection infrastructures and clouds.

Eduroam is a successful confederation joining Service- and Identity Providers as worldwide participants providing WiFi Internet Access that a student can trust to work hassle free. We have shown that several elements of the eduroam Compliance Statement do fit the essence of the SPG framework.

6 Conclusions

"When I am working on a problem, I never think about beauty but when I have finished, if the solution is not beautiful, I know it is wrong."

R. Buckminster Fuller (1895-1983)
Architect, systems theorist, author, designer and inventor

6 Conclusions

In this thesis we have investigated what is needed to provide policy based authorized access to networked resources that are owned by multiple autonomous parties that have to trust each other to provide a service across domains. We have done this by using the AAA Authorization Framework and Generic AAA Architecture described in chapter 2 as a starting point.

The Framework introduces a way to describe three fundamental sequences between a User, a Service and an AAA Server as the Authority taking policy decisions about resources belonging to a Service Provider. In this thesis we have considered the Agent Sequence and Push Sequence and a combination of both and shown its applicability using the AAA Architecture Research Group work as a starting point. We used functions defined in the Generic AAA Architecture to implement AAA Servers in a network to take policy decisions in a distributed way. Key to the Generic AAA Architecture is that we separate the "if-then-else constructs" based driving policy handling from the application specific semantics involved. The architecture defines a Rule Base Engine (RBE) function to perform the logic of handling a driving policy and an Application Specific Module (ASM) concept to handle the attributes representing the semantics of the policy evaluation and its decision. An ASM function could interface to the outside world in various roles, for example: Provision equipment with information to allow policy enforcement, act as Policy Enforcement Point, act as user sending requests to another domain to request resources, interface with a resource manager or other services controlling resources and more. In this way, the Generic AAA architecture is able to support various combinations of Authorization Framework sequences.

By means of studies and demonstrations of chapters 3 and 4 we have investigated and validated ways to implement Generic AAA architecture functions acting according to sequence models defined in the Authorization Framework. Members of the SNE group at UvA created a Generic AAA Toolkit using JAVA in a J2EE application server environment to experiment with various scenarios in the context of high performance optical networking. Here applications, with envisaged use in e-Science environments, have a need for on-demand or scheduled use for dedicated network resources. The experiments with the Generic AAA toolkit and its ways to implement Authorization Framework sequences focussed on the control of such resources, although the use of the Generic AAA toolkit concepts could be more general.

To deliver a global solution for e-Science communities, autonomous National Research and Education Network providers must collaborate to provide dedicated high-bandwidth connectivity using optical technology as part of an e-Infrastructure. As authorization transactions cannot take place before some understanding has been established amongst involved parties, we have studied what it means to establish such understanding. This understanding allows parties to trust each other when providing an end-to-end service. By considering existing cases from the payment card industry and the higher education WiFi roaming world, we have derived a framework describing Service Provider Groups. This framework, described in chapter 5, provides a way to think about creating such groups where the number of actors involved and impact of risk helps

determine if the personal- or impersonal form of trust or power should play a predominant role.

In cases where service providers collaborate, it is important to understand that trust is built by ensuring that all involved parties have the knowledge to understand that the policies a party uses to authorize and handle a transaction are correct. This correctness is ensured by administering applicable group rules and have the enforcement in place to correct any mistakes. In this way a party will earn trust from other parties.

Renting a car is typically authorized by a credit card transaction. Such transaction is authorized by executing policy rules at the cardholders bank checking if the account is in good order and the payment limit is not exceeded. Both the car rental company and the cardholder trust the MasterCard system to handle such transaction in the correct way. MasterCard ensures that all involved parties will handle this transaction in the correct way by administering and enforcing their rules. Each party will translate these rules into policies operating the authorization system. Although not driven by payments and the involved financial risk, e-Science communities can benefit from the way of thinking behind the construction of a credit card authorization system as it also allows autonomous (and competitive) banks to collaborate. We have seen that MasterCard employs three kinds of powers: Rulemaking, Judicial and Executive power. Such powers must be represented in one form or the other, in particular when the amount of parties and involved risk scales up. An authorization system built without such foundation can typically not go beyond serving small communities. We have seen that eduroam is good example able to scale up globally because it arranged such powers via a confederation.

The studies performed in this thesis can answer the research questions posed in chapter 1 to find out what is needed.

6.1 Generic Authorization Functions

What generic authorization functions can be distinguished and how do they interact?

In chapter 2 we have explained that there are essentially three parties and an optional fourth party that interact to perform authorization functions. The original definitions established during the work in the IETF were refined in the work at the OGF. We distinguished:

1. **The User or Subject:** An entity (e. g. a person or process) that can request, receive, own, transfer, present or delegate an electronic authorization as to exercise a certain right.
2. **The User Home Organization:** The (optional) Organization that administers a user by determining and providing attributes that describe a User (e.g. access rights, quota, roles, etc.) that may be evaluated during a policy decision.
3. **The Authorization Authority or AAA Server:** An administrative entity that is capable of and authoritative for issuing, validating and revoking an electronic means of proof such that the named subject (a.k.a. holder) of the issued electronic means is authorized to exercise

a certain right or assert a certain attribute. Right(s) may be implicitly or explicitly present in the electronic proof.

4. **The Service Equipment or Resource.** The entity that represents the service, which needs information that authorizes the usage of the service offered by the equipment. A component of the system that provides or hosts services and may enforce access to these services based on a set of rules and policies defined by entities that are authoritative for the particular resource.

In the AAA Authorization Framework, we recognized that above entities can relate and interact in a number of different sequence models, most importantly in the following three sequence models:

- **The agent sequence model**, where the user sends a request to the authority that provisions the service. The agent will reply to the user if it could honour its request and provisions the service to admit the user.
- **The pull sequence model**, where the user sends a request to the service. The service outsources the decision to provide access to the authority. The authority will decide if the user can be admitted.
- **The push sequence model**, where the user sends a request to the authority. The authority decides to admit the user it will reply with an authentic object (e.g. a token) that can be recognized by the service to admit the user.

During our subsequent research we distinguished the combination of the agent and push sequence model as the token model. Here the provisioning of the service by the authority with the meaning of a token is an explicit phase.

We have seen that the above entities and sequence models are able to describe multi-domain authorization scenarios with multiple authorities and multiple services. An optional "User Home Organisation", carrying knowledge about users from a certain domain, can be useful when describing roaming scenarios where such organisations carry attributes of users that can be used to take authorization decisions next to information that allows Authentication.

We have seen that a AAA server, build using the Generic AAA architecture, can function as the authority in multi-domain scenarios. The Generic AAA architecture separates the logic of decision taking from the semantic handling of a decision. This architecture was built around the idea that multi-domain scenarios are expected to benefit from simple communication of "yes or no" decisions upon requests as those decisions can be easily combined. Any attribute information is handled transparently by the protocol and decision taking mechanism.

The Generic AAA architecture distinguishes the following main functions that we studied and found essential:

1. **A Rule Based Engine** to perform the logical handling of policy decisions based on a simple if-then-else construct.
2. **A policy repository** containing driving policies that are addressed by requests.
3. **An Application Specific Module** handling the semantic elements defined by a policy and is able to interface this meaning to the outside world.

The architecture describes that Rule Based Engines and Application Specific Modules can be build allowing policy based distributed decision making to be performed by communicating authorization messages between these elements using various types of protocols and message objects. The architecture does not assume any particular protocol that should handle message objects. Application Specific modules understand the meaning of message objects, and are called by a Rule Based Engine executing a driving policy that determines the logic of handling an authorization transaction. A network of interacting Generic AAA servers can be built to allow distributed decision taking where AAA servers may split authorization request messages into relevant parts for particular domains and subsequently combine decision results and have the ability to communicate the decisions via ASM's into meaningful results that provision services, manage resources, issue access tokens, etc.

6.2 Authorization Concepts

What generic authorization concepts are expected to work best for classes of applications that use multi-domain network resources?

In chapter 3 we have described the application of the Agent model, the Push model and its combination. The pull model can be observed to work well in roaming scenarios. Such can be seen from the eduroam example described in chapter 5. The pull model is used in this case by a single service domain, requesting access authorization for users belonging to multiple (home-)domains. In this scenario, however, having such domain service become responsible for the coordination of services offered by multiple other domains seemed less feasible. This initial observation is the reason why we studied application of the Agent model first. During the initial experiments we discovered that chaining requests via a number of Agents could be a potential bottleneck for obtaining fast responses to requests in multi-domain scenarios. In particular the technology handling and parsing XML messages, was showing poor performance as we saw with **our** experiments from SuperComputing 2004.

As in many Grid style e-Science applications, workloads are scheduled anyway, we decided to further study the application of the Push model. The Push model allows the separation in time of the handling of a request and enforcing the access request in the service. A scheduling process can make sure it has obtained resources before allowing applications to use them at a later (scheduled) point in time. Moreover, by handing the proof of authorization (e.g. a token) to the application, it is possible to allow specific applications to use a resource (and not others) and therefore allows more granular control.

Tokens can potentially be inserted at various functional layers of the network technology and this was not a common approach. This was the reason for deciding to experiment more with the applicability of this idea, assuming that creating a token and providing the meaning of a token to the a service could still be performed using the Agent model. This would also allow each domain to attach its own meaning (according to its own policies) to be attached to a common token that is communicated from one domain to another to obtain access to a domain resource. This is why the token model was recognized as the most promising concept to be further explored in these type of scenarios. The token model became a main subject of our experiments.

6.3 Multi-domain Authorization Applications

How can we apply the generic multi-domain authorization concepts in Network QoS / Lightpath provisioning class of applications?

We conducted and described seven experiments that demonstrated how the Generic AAA concepts can be applied using the Agent- and Push sequence model and its combination, the Token model:

1. iGrid 2002: Authorization of a single-domain QoS Path based on Generic AAA
2. Supercomputing 2004: Agent model drives the creation of a multi-domain lightpath
3. GridNets 2005: Switching IP packets based on token recognition.
4. iGrid 2005: Token based lightpath access authorized by a service plane.
5. SuperComputing 2005: Token based lightpath access authorized by a service plane.
6. SuperComputing 2006: Token based lightpath access authorized by a control plane (single-domain scenario).
7. SuperComputing 2007:Token based lightpath access authorized by a control plan (multi-domain scenario)

In these experiments we have been able to demonstrate:

Experiment 1 (section 4.1) conducted at iGrid2002: Building a single domain QoS path using a pair of VLAN switches interconnected via a 1 GB optical connection. This setup was using the full control model described in section 3.1.3, where the controlling entity was implemented using a Generic AAA server using the Agent sequence. We demonstrated how a Rule Based Engine was executing a Driving Policy upon receiving a request, involving multiple Application Specific Modules to control the VLAN switches and to perform resource management for the 1 GB connection. This experiment showed the viability of the Generic AAA concept as basis for further experiments.

Experiment 2 (section 4.2) conducted at Supercomputing 2004: Here we showed how an application can drive the creation of a lightpath across multiple domains, allowing our Generic AAA server concept to authorize network lightpath elements established by a network

provisioning system called DRAC from Nortel Networks. Combinations of AAA servers and DRAC agents build a Service Plane controlling multiple network domains. During the demonstration we showed our ability to create a lightpath across three domains (parts of the NetherLight, Starlight and OMNINet testbed), where each domain provided a part of an end-to-end lightpath. DRAC/Generic AAA server agents controlled each of its inter-domain links. We showed how XML requests can be build and subsequently handled by a Rule Based Engine. Here we saw semantic (information needed to be passed/retrieved to/from the DRAC agents) being handled transparently by Driving Policies executed by Rule Based Engines and Application Specific Modules (ASM) of each AAA server. We saw the Driving Policy interact via ASMs with an Authentication Service, a service to have a DRAC provision a lightpath, allow route determination to the next domain, manage a session and prepare for accounting. The AAA servers acted in the Agent model that formed a chain along the path to be provisioned. The next link to be used was determined by a particular DRAC agent. The network was build using redundant connections. We saw DRAC signal a path failure, upon which the Driving Policy took action to initiate the provisioning of an alternate link. Measurements were performed on the failover times, as it required similar actions to setup a new connection. Although significantly faster than manual failover procedures, this experiment showed the short comings of the Agent model in terms of time (± 75 sec.) required to setup or failover a connection.

Experiment 3 (section 4.3+4.5.2.1) was conducted at GridNets 2005. This experiment was the first in the series looking at using tokens to authorize path access. Being concerned about the time required to setup a connection at the time a connection is needed when using the Agent model, we started to experiment with tokens as a means to provide authorized access considering also the arguments discussed in section 6.2.
In this experiment we showed the use of tokens inside an application flow of IP packets that are enforced by a network switch device that would select a particular authorized (pre-setup) path instead of a default path if tokens were recognized as valid. An Intel IXDP 2850 Network Processor development platform served as switch, micro-programmed to create and recognize a token embedded in the IP Options field. We argued that this switch could be used at an inter-connection point between a hybrid network and a lambda grid to perform real time path selection. As, according to the original IP standard, a router should forward packets with IP options unmodified, this approach allows flows containing tokens be transparently handled by connectionless networks. According to the device specifications, the used platform was expected to handle flows at speeds of up to 10 GB/s.

Experiment 4+5 (section 4.4+4.5.2.3.1) were conducted at iGrid 2005 and SuperComputing 2005. These experiments were performed to explore the use of the token model to authorize an application to access a path that was provisioned by Nortel's DRAC, acting as service plane, in a single domain case. Issuing a token was a process performed by a Generic AAA server after contacting DRAC to request the provisioning of a link at a particular time and subsequent creation of a token. This token allowed access enforcement to this link using an optical switch in front of the connection. The optical switch was driven by a Generic AAA server that recognized the token and its meaning. The application used was the Virtual Machine turntable experiment

that circled a VM across three different locations using a high capacity dedicated network connection between the locations. The iGrid 2005 experiment measured the times needed to request a connection and issue a token and the time required to provide access to the link and become operational using the token. Issueing and verifying a token by the Generic AAA server could be performed at acceptable speed (<100 ms). DRAC was capable of setting up a connection in around 10 seconds, however such connections could be provisioned ahead of usage time.
Main concern was the time necessary to propagate the link-up state inherently caused by use of a photonic switch. Based on the experiments, we conclude that the token approach was a promising way forward.

Experiment 6 (section 4.5.2.2) was conducted at SuperComputing 2006 together with the team lead by Internet2 involving ESnet. It showed how a token can be inserted into the Policy_Data object of an RSVP-TE message that was used as a control plane signalling mechanism of an Virtual Label Switch Router (VLSR) developed by the open-source GMPLS DRAGON project. A Generic AAA server was used to create a single-domain case to proof its concept. The Generic AAA server interacted with both a VLSR and a scheduling application to act as resource manager. This experiment proved the applicability of the token concept and motivated further experiments with Internet2. The token concept became part of experiments with the Internet2 DCN project.

Experiment 7 (section 4.5.2.3.2) was conducted at SuperComputing 2007 together with the Internet2 DCN project, Nortel Networks and ESnet. Here we build a multi-domain case, using a Token Validation Service (TVS) as Generic AAA Toolkit component. The TVS was integrated into the Internet2 developed InterDomain Controller acting as Lightpath Authority together with a Token based Policy Enforcement Point integrated into the DRAGON VLSR. The setup was capable of generating tokens during the reservation phase of a path and subsequent enforcement during the handling of an RSVP-TE PATH message at GMPLS control plane level to perform authorized establishment of the lightpath.

Overall, the token-based experiments (4, 5, 6 and 7) show that at a token can be applied as an abstract and shared representation of a permission to access one or more service instances. The token is obtained by applying the Token Sequence model by having the User first contact an Authority to obtain a token and have the Authority provision the Service. Subsequently, the User presents the token to the Service. Here, the token is presented as part of an access request to a service. Multiple Authorities and Services can be chained. A service request can be forwarded to by one Authority to a next Authority that can arrange part of the service chain. Together, authorities create a token that can be recognized as proof of access authorization by all participating service domains. When presented to a service, the token represents some form of abstract, authentic and integrity-checked index, pointing to a pre-allocated service instance that was defined during the authorization phase by each individual authority (AAA server) acting as Service Agent. The service is subsequently instantiated and made uniquely accessible inside the Service Equipment by information (service parameters, key material, etc.) provisioned by

its Service Agent. Note that each individual domain can use its own policies to create such an instance, as long as the policies adhere to the rules that define and govern the delivery of an end-to-end service.

We have seen that a token can be applied at different levels at the network: At lower IP level, at Control Plane level using for example RSVP-TE or at Service Level. When used at IP level, enforcement can be very granular without the need to have unique application access to the end points of a connection to enforce application level access.

The application of the Token model, where AAA Agents act as "Lightpath Authority", has proven to be suitable to coordinate resource management and to provisioning the right information. The model was successfully used in examples used with the Internet2 DCN experiment and is part of the approach used by the OGF's NSI Working Group defining Network Service Agents (section 5.1.1).

The pull model could in theory also be used, however its feasibility has not been studied and seems at first less applicable when there is a need to coordinate resources across domains. We expect that the Pull model is better suited to have a single domain contact multiple independent User Home Organisations, which registers information about individual users as described in the Eduroam case (section 5.3.2). Its suitability to arrange a chain of services across multiple domains is left for further research.

How correct policies, determining the behaviour of a domain, can be established such that a domain can be trusted by the community, is subject of our last research sub-question.

6.4 Arranging Trust

What is needed to arrange trust when authorizing e-infrastructure resources?

In chapter 5 we defined the concept of a Service Provider Group as a group of member organisations that act together as a business providing one or more services that none of its members could provide on its own. Users and service providers need to have a willingness to rely on each other when services are delivered. The willingness to rely on something or somebody is an understanding associated with trust. In our initial AAA Authorization framework study we recognized that "Trust is necessary to allow each entity to "know" that the policy it is authorizing is correct".

We stated that trust is a broad concept studied in areas such as sociology and psychology. Therefore, we first need to define what trust means within the organisational context. Here, different actors need to have relationships that coordinate business activities delivering goods or services. We used the concepts and studies by Nootenboom and Bachman to extract definitions usable within this context.

We started with the observation that trust in organisational context is predominantly considered as *"a means to cope with uncertainty"*. Trust inherently introduces a risk, as trust can be disappointed. Knowledge about applicable rules and the potential of sanctioning provide *good reasons* that are considered effective ways to confine risk of disappointment. Such knowledge can be tacit, an understanding living in the minds of people interacting to create the basis for *personal trust*. This type of trust works for relative small communities. When communities scale up, knowledge about rules must be made explicit. i.e. written down in an *impersonalized form* such that it can be shared and become workable for larger communities.

We also recognized the role of power as an additional means to provide *good reasons* to confine risk of disappointment. Power is only meaningful if there is a realistic threat of sanctions. Relationships are often based on a combination of trust and power. If the impact of risk increases, relationships tend to rely more on power. We showed that power and trust exist both in personal and impersonal forms. Bachmann argues that in large, strongly regulated organisations impersonal power and trust tend to link into each other in such a way that powerful intra-organisational and environmental structures breed trust between individual actors in a highly efficient manner. In weakly regulated organisations, individual efforts to establish cooperation between relevant actors in the organisation becomes more important. Based on this we created a model relating the number of actors and impact of risk, showing a diagonal axis along which the role of a Service Provider Group could be seen as a way to help organisations move upwards as they scale up and more impact of risk is involved.

We found MasterCard as an example fitting the extreme top end of the axis of the aforementioned model. MasterCard is an organisation operating at worldwide scale with banks as a group of autonomous and competitive Service Providers. This group of service providers can handle card payment authorization transactions via a network involving its members to handle the transactions that is trusted to initiate the transfer of money from a cardholders account into a merchants account. We examined the MasterCard rules to recognize a framework for a strongly regulated organisation. Inspired by the principles of the "Trias Politica" we recognized from its rules that MasterCard possesses three essential powers constituting its service provider group: Rulemaking-, Judicial- and Executive Power (performing administration and enforcement). In this way MasterCard is able to provide the impersonal power needed to conduce trust amongst its service providers such that each involved entity knows that the policies it is authorizing are correct. This is the essential role of MasterCard, which allows its service to be delivered in such way that all involved members and customers are willing to rely on it.

Using the above framework we conceptualized a framework for a Service Provider Group. The framework recognizes the need for a SPG Governing Organisation and Provider/user organisation. Each organisation recognizes a Directorate holding the three powers needed in principle to create a strongly regulated organisation. It depends on how strongly the need is to regulate (depending on the number of actors and impact of risk) how much impersonal power needs to be implemented.

Part of an organisation is "humanly managed" and the service providing part is "protocol managed". The humanly managed part administers and enforces the protocol managed part. The SPG Governing organisation is needed to establish common rules that govern the delivery of a service by the provider organisation or correct administration of users. In Provider/User organisations the Administration essentially translates group rules, its own rules and possibly national laws into policies that are executed by the service delivering part. This way of thinking allows each service provider to implement a service based on common rules, but according to its own policies. The administration of rules is needed to determine that a Service Provider implements the correct policies to provide a service. The role of the enforcement is needed to ensure the correct execution of the policies.

To answer our research question, we must place the above in the e-Infrastructure context. In this context, the Internet enables research that is increasingly carried out through distributed regional, national and global collaborations. Typically collaborating research communities start small where authorizing access to community resources is typically build on personal trust. Considering the Optical Networking context, this is the way how the Global Lambda Infrastructure Facility (GLIF) community emerged. Directors of National Research and Education Networks, agreed to contribute their spare optical network capacity to this common initiative by signing a single page agreement. This allowed maximum freedom in exploring potential capabilities of the underlying infrastructure technologies and ways to control them. However, if such initiative has to scale into a global infrastructure, capable of being used by a much larger educational community, the Service Provider Group framework is a way to think about what is needed in terms of (additional) authorization mechanisms and organisation elements arranging policies needed at operational level.

6.5 Main question

We asked ourselves the main question:

What generic authorization functions are needed to provide trusted, policy based access to combinations of e-Infrastructure resources that are owned by different parties?

The example in the introduction described an early system, authorizing transactions around a common pool of resources shared by Neolithic communities that arranged contribution and re-distribution. Here we saw the importance of a need to create community rules to manage contribution and access to resources a system implementing its rules by using clay tokens symbolizing amounts of resources and envelopes symbolizing a transaction. Protocols between humans and policies are likely to have handled the process that arranged such outcome. As writing was not invented yet (preventing rules to be made explicit) the correct handling of such policies was based on personal trust and personal power of the actors involved in this process. People handling the authorization system were expected to have correct knowledge about policies to handle tokens. This fact created the necessary trust in the system.

In modern digital systems, providing authorized access to network resources, the principle of having the correct knowledge about policies to handle authorization transactions still applies. When combining different network resources, owned by different parties that together provide an end-to-end network service, there is a need to clearly define what this network service is. It is also necessary to provide the correct knowledge to each party such that it can be trusted to provide its contribution in the expected way. Defining the service and rules each contributing party should apply is something that needs to be arranged in common. We defined the Service Provider Group (SPG) as a framework to identify functions of technical- and organisational elements.

The organisational elements define, administer, enforce and judge the rules, all under the responsibility of a SPG directorate. Once the common rules are defined, they should be combined with a SPG member's own rules to ensure its desired autonomy. National law and regulations may provide additional context. As such, a contributing member defines its policies that must be executed and enforced to provide and allow authorized access to its service. In our framework, a party is trusted if it uses policies that are known to be correct to authorize its contribution such that they are compliant to the rules defined by the group in an enforceable way.

In modern technical systems, next to being manually handled by operators, policies are executed in an automated way and use secure protocols and message objects to communicate. Next to providing the correct policies to the technical systems, based on common rules arranged by the organisational elements, the correct messages and protocols must be defined to allow communication between SPG members according to a commonly understood meaning. We identified the actors and message sequences without making assumptions about protocols and

message objects that should be exchanged in an effort to maintain generality of our approach. We found that the Generic AAA architecture was an applicable approach by implementing its concepts in our Generic AAA toolkit and JAVA programs. Specific protocols could be handled by implementing Application Specific Modules, without losing the generic character of our approach.

We showed the ability of Generic AAA servers to handle authorization transactions in a distributed way. We saw AAA servers executing driving policies that combined resources into an end-to-end service using different models of Authorization framework: the Agent- and Push model and its combination. As such we found that exchanging tokens as authentic and integrity secured object, was a promising way forward to arrange multi-domain authorization of resources. A token was used as a pointer to a service instance and was used in the combined Agent- and Push model (that we called the Token Model). The approach allowed each domain to pre-arrange resource access in an autonomous way before a token is used as a way to access the service. Agents can pre-arrange the setup of resources using policies that could be arranged via a Service Provider Group. We concluded that we expect that the more SPG rules, translated into domain policies, define and abstract a service to be delivered, the less complexity (number of attributes and need to align its meta-data) is needed in objects exchanged between domains. This design principle keeps resource access enforcement with tokens as efficient and simple as possible.

7 Future directions

"The past, like the future, is indefinite and exists only as a spectrum of possibilities."

Stephen Hawking (1942)
Theoretical physicist and cosmologist

7 Future directions

We started our research with the understanding that resource owners, representing a domain, have individual stakes, concerns, rights, obligations, etc. when contributing their resources as part of an end-to-end service. Arranging automated access to such resource chains must take these concerns into account. Such a requirement creates the need for a multi-domain authorisation mechanism, capable of handling authorization transactions in a way that is trusted amongst community members. Trust emerges from the correct knowledge each domain has when executing policies. This knowledge is based on group rules, which are translated into policies that the operational layer uses to handle transactions. Our research into the "multi-kingdom problem" resulted in an authorization mechanism based on simple tokens. A simple token is used as a reference to a service when presented at each domain. The referred service element may have been (pre-) arranged in different ways by each domain using a common understanding of the correctness of the service to be delivered. The presented Service Provider Group Framework helps in the understanding of what is needed to allow trust to emerge. Here, each owner has the freedom to autonomously determine their policies, whilst understanding the need to be correct in delivering a service.

We have shown a high level framework describing Authorization sequences, supported by a Generic Architecture, and a framework to create Service Provider Groups. The frameworks and Architecture certainly needs more detailed understanding. We have started to understand the applicability of our Generic AAA approach. In the area of SPGs, future work is needed to study if other cases fit the framework, not only e-Infrastructure cases.

At the start of our research we did foresee a scenario where a user could combine the delivery of an online movie, the required network bandwidth and a pizza as something that has only value if these three items can be combined. We saw how our Generic AAA approach could solve this problem technically. In chapter 5 we considered a theoretical approach to administer and provision policies in such a way that the outcome of authorization decisions should be monitored and enforced, which provides good reasons such that its correctness can be trusted. Additional research is however needed to verify the applicability of this approach, in particular in cases where IT infrastructures and application services are becoming more and more software definable.

For both for the e-Science- as well as the Enterprise domain, we see a number of ways to continue this research, Within the e-Science domain, we see that our research can continue to be applied to Optical Networking, in particular within the work on the Network Service Interface as being defined by the OGF NSI working group. Here our SPG concept can help in setting up a Global organisation that can provide lightpath services to much larger communities at worldwide scale within the GLIF context.

In many Enterprise e-Commerce scenarios, the Internet increasingly provides services that are offered via API's instead of browser based interfaces. API's are available to book travel [EXPE],

arrange a taxi [UBER], find a place to stay [AIRBNB], etc. Both end users and businesses (e.g. acting as brokers that are willing to take risk) are increasingly developing mobile apps, which combine such services, sometimes in unforeseen and very competitive ways,. As the business logic (implemented by browser-based web applications) have largely disappeared when services become accessible via APIs, the way policies govern service access may need to be reconsidered and possibly become replaced by other, more distributed policies decision taking mechanisms. We might explore the feasibility of creating a Service Provider Group for broker functions. Brokers that are trusted to combine services in a predictable way managing the involved risks.

The token concepts have also been introduced in the work of User Programmable Virtualized Networks (UPVN) [MEIJ] and work that emerged from this [CRIS9, STRI]. Instead of providing correct policies, a Service Provider Group could also provide the correct programs in a UPVN infrastructure.

Addressing the complexity to arrange security, trust, and access authorization to ICT Infrastructures in general has since the start of our research become a mature field of research [NCSRA]. In a world, where the number of resources, participants and potential relationship complexity increases, a scalable approach to problems surrounding trust and authorisation needs further investigation.

By using the simple token based mechanisms, the author images that the Internet would ultimately be able to offer "Business Class" services allowing businesses to work together with Internet Service Providers to deliver special quality services (e.g. guaranteed network bandwidth, security, always available) to distinguish themselves from the competition. Although progress has been made at present day by for example services offered by content delivery networks [AKAM], enterprises cannot hand simple tokens to individual customers. Such tokens would enable a different user experience in network quality when for example performing electronic banking, check-in for their flight or change a hotel reservation even if a site is under DDoS attack. We show that traffic, containing such tokens, can be routed and treated differently by each domain.

In the area of Enterprise domain security, our Service Provider Group approach could help define new ways to provide security by creating a collaboration between Enterprises and Internet Service Providers that could provide end-to-end security services to their mutual customers. Further research into the application SPG concept will be part of a National NWO research project called SARNET [SARN] that is expected to start early 2015.

Scientific contributions by the author

8

"To effectively communicate, we must realize that we are all different in the way we perceive the world and use this understanding as a guide to our communication with others."

Tony Robbins (1960)
Life coach, author and motivational speaker

8 Scientific contributions by the author

Google scholar shows that the author has accumulated 1232 citations to date with an h-index [15] of 16 and an i10 index[16] of 25.(Oct 1st 2014). These numbers are considered relatively high for the engineering community in which this work took place.

8.1 Co-author contributions to IETF Documents of chapter 2

RFC2903 "Generic AAA Architecture", C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, D. Spence, IETF August 2000.

Apart from actively discussing its architecture with the co-authors and contributing to modelling the bandwidth broker, e-commerce, mobile-ip and education & distance learning use cases [R2905] that helped to build RFC2904 and this RFC, my particular contribution was to help think along the lines of separating the logic of decision taking from the semantic handling of a decision. My expectation was that it would make the process of decision taking in multi-domain scenario's easier. Rather than communicating complex sets of attributes, simple binary decisions communicated between domains, (were each domain can autonomously take decisions on a commonly understood service) was my envisioned way multi-domain authorization could work. Combining the concepts of the RAP Working Group [RAPWG, RAJY] by envisaging a multi-domain network of Policy Decision Points "RADIUS proxy style" [R2607] (inspired by John Vollbrecht's work) helped me push idea's around how a such network could be constructed. Combining these thoughts with other AAA Research Group members on ways to implement these thought in a layered architecture (in particular Cees de Laat, John Vollbrecht, Arie Taal and Dave Spence), lead to the definition of the Rule Based Engine and Application Specific Module concepts defined in this RFC.

RFC2904 "AAA Authorization Framework", J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, IETF August 2000.

My main part was (in close collaboration with John Vollbrecht) conceptualising the AAA Framework entities and sequence models, which formed the foundation of this RFC and foundation of my future work into researching the applicability of these concepts. Also the recognition that multi-domain authorization must be considered by combining both the protocol issues with the business issues has been a major driver for me resulting in chapter 5.

GFD-I.038 "Conceptual Grid Authorization Framework and Classification", M. Lorch, B. Cowles, R. Baker, L. Gommans, P. Madsen, A. McNab, L. Ramakrishnan, K. Sankar, D. Skow, M. Thompson, OGF November 2004.

---

15 h-index is the largest number h such that h publications have at least h citations.

16 i10-index is the number of publications with at least 10 citations.

In this working group I helped to clarify an sharpen the definitions from the RFC2904 work as included in chapter 2.

Also, in support of chapter 2, the following documents were published to show the applicability of the Generic AAA Framework and Architecture and some essential requirements:

RFC2905 "AAA Authorization Application Examples". J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, IETF, Aug. 2000.

RFC2906 "AAA Authorization Requirements", S. Farrell, J. Vollbrecht, P. Calhoun, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, IETF, Aug. 2000.

8.2 Lead author publications used for chapters 3 and 4

Leon Gommans, Cees de Laat, Bas van Oudenaarde, Arie Taal, **"Authorization of a QoS Path based on Generic AAA"**, iGrid2002 special issue, Future Generation Computer Systems, volume 19 issue 6, pp. 1009-1016 (2003) DOI: 10.1016/S0167-739X(03)00078-5.

Leon Gommans, Franco Travostino, John Vollbrecht, Cees de Laat, Robert Meijer, **"Token-based authorization of Connection Oriented Network resources"**, GRIDNETS conference proceedings, oct 2004.

Leon Gommans, Cees de Laat, Bas van Oudenaarde, Arie Taal, **"Authorization of a QoS Path based on Generic AAA"**, iGrid2002 special issue, Future Generation Computer Systems, volume 19 issue 6, pp. 1009-1016 (2003) DOI: 10.1016/S0167-739X(03)00078-5.

Leon Gommans, Franco Travostino, John Vollbrecht, Cees de Laat, Robert Meijer, **"Token-based authorization of Connection Oriented Network resources"**, GRIDNETS conference proceedings, oct 2004.

Leon Gommans, Cees de Laat, Robert Meijer, **"Token Based path authorization at Interconnection Points between Hybrid Networks and a Lambda Grid"**, IEEE GRIDNETS2005 proceedings, ISBN 0-7803-9277-9. DOI: 10.1109/ICBN.2005.1589768 © 2005 IEEE *.

Leon Gommans, Bas van Oudenaarde, Freek Dijkstra, Cees de Laat, Tal Lavian, Inder Monga, Arie Taal, Franco Travostino, Alfred Wan, **"Applications Drive Secure Lightpath Creation across Heterogeneous Domains"**, IEEE Communications Magazine, Feature topic Optical Control Planes for Grid Networks: Opportunities, Challenges and the Vision, vol. 44, no. 3, March 2006, DOI: 10.1109/MCOM.2006.1607872 © 2006 IEEE *

L. Gommans, B. van Oudenaarde, A. Wan, C.T.A.M. de Laat, R. Meijer, F. Travostino and I. Monga, **"Token Based Networking: Experiment NL101"**, iGrid2005 special issue, Future Generation Computer Systems, volume 22 issue 8, pp. 1025-1031 (2006). DOI: 10.1016/j.future.2006.03.025.

Leon Gommans, Li Xu, Fred Wan, Yuri Demchenko, Mihai Cristea, Robert Meijer, Cees de Laat , **"Multi-Domain Lightpath Authorization using Tokens",** Future Generation Computing Systems, Vol 25, issue 2, 2008, pp 153-160, DOI 10.1016/j.future.2008.07.013.

8.3 Lead author publication used for chapter 5

Leon Gommans, John Vollbrecht, Betty Gommans, Cees de Laat, **"The Service Provider Group Framework"**, Future Generation Computer Systems. DOI: 10.1016/j.future.2014.06.002.

8.4 Co-Author papers, directly related

S.M.C.M. van Oudenaarde, Z.W. Hendrikse, F. Dijkstra, L.H.M. Gommans, C.T.A.M. de Laat, R.J. Meijer, **"An Open Grid Services Architecture Based Prototype for Managing End-to-End Fiber Optic Connections in a Multi-Domain Network",** High-Speed Networks and Services for Data-Intensive Grids: the DataTAG Project, special issue, Future Generation Computer Systems, volume 21 issue 4, pp. 539-548 (2005).

F. Travostino, P. Daspit, L. Gommans, C. Jog, C.T.A.M. de Laat, J. Mambretti, I. Monga, B. van Oudenaarde, S. Raghunath and P.Y. Wang, **"Seamless Live Migration of Virtual Machines over the MAN/WAN",** iGrid2005 special issue, Future Generation Computer Systems, volume 22 issue 8, pp. 901-907 (2006).

M. Cristea, R. Strijkers, D. Marchal, L. Gommans, R. Meijer, C. de Laat., **"Supporting Communities in Programmable Grid Networks: gTBN",** proceedings of the FIP/IEEE International Symposium on Integrated Network Management, 2009, page 406 - 413.

8.5 Co-author journal papers helping related research

B.U. Niderost, L. Gommans, G. Kemmerling, M. Korten, C.T.A.M. de Laat, W. Lourens and E.A. van der Meer, **"Objectivity / Corba Distributed Database Performance on a Gigabit Sun-ultra-10 Cluster",** IEEE Trans. on Nuclear Science, April 2000, vol.47, nr.2, p313.

8.6 Co-author conference papers helping related research

Matias, J., E. Jacob, Y. Demchenko, C. de Laat, L. Gommans, **"Extending AAA Operational Model for Profile-based Access Control in Ethernet-based Neutral Access Networks",** Proc.The First International Conferences on Access Networks, Services and Technologies (ACCESS 2010), September 20-25, 2010, Valencia, Spain. Pp. 168-173.

Y. Demchenko and L. Gommans and C.T.A.M. de Laat, **"Extending role based access control model for distributed multidomain applications",** IFIP International Federation for Information Processing, 2008, Volume 232, pages 301-312.

Freek Dijkstra, Bas van Oudenaarde, Bert Andree, Leon Gommans, Paola Grosso, Jeroen van der Ham, Karst Koymans and Cees de Laat, **"A Terminology for Control Models at Optical Exchanges",** LCNS, Volume 4543, july 2007, Page 49-60.

Yuri Demchenko, Frank Siebenlist, Leon Gommans, Cees de Laat, David Groep, Oscar Koeroo, **"Security and Dynamics in Customer Controlled Virtual Workspace Organisation",** Proceedings of the 16th international symposium on High performance distributed computing, Monterey Bay California, June 2007, page 231 – 232.

Yuri Demchenko, Leon Gommans, Cees de Laat, **"Extending User-Controlled Security Domain with TPM/TCG in Grid-based Virtual Collaborative Environment",** In Proceedings The 2007 International Symposium on Collaborative Technologies and Systems (CTS 2007), May 21-25, 2007, Orlando, FL, USA. ISBN: 0-9785699-1-1.

Demchenko Y., L. Gommans, C. de Laat, **"Using SAML and XACML for Complex Authorisation Scenarios in Dynamic Resource Provisioning",** The Second International Conference on Availability, Reliability and Security (ARES 2007), Vienna, April 2007, proceedings, page 254-262.

Robert J. Meijer, Rudolf J. Strijkers, Leon Gommans, Cees de Laat, **"User Programmable Virtualized Networks"** proceedings of The 2nd IEEE International Conference on e-Science and Grid Computing, Amsterdam, Dec 2006, ISBN: 0-7695-2734-5, page 43.

Yuri Demchenko, Leon Gommans, Cees de Laat, Rene van Buuren, **"Domain Based Access Control Model for Distributed Collaborative Applications",** proceedings of The 2nd IEEE International Conference on e-Science and Grid Computing, Amsterdam, Dec 2006, ISBN: 0-7695-2734-5, page 24

Demchenko, Y., L. Gommans, C. de Laat, A. Taal, A. Wan, O. Mulmo, **"Using Workflow for Dynamic Security Context Management in Complex Resource Provisioning"**, 7th IEEE/ACM International Conference on Grid Computing (Grid2006), Barcelona, September 28-30, 2006. IEEE Cat. No. 06EX1363C. ISBN: 1-4244-0344-8, pp.72-79.

Demchenko, Y., L. Gommans, C. de Laat, A. Tokmakoff, R. van Buuren, **"Policy Based Access Control in Dynamic Grid-based Collaborative Environment"**, The 2006 International Symposium on Collaborative Technologies and Systems, Las Vegas, May 14-18, 2006, Proceedings. IEEE Computer Society. ISBN: 0-9785699-0-3, pp. 64-73.

Demchenko, Y., L. Gommans, C. de Laat, M. Steenbakkers, V. Ciaschini, V. Venturi, **"VO-based Dynamic Security Associations in Collaborative Grid Environment",** The 2006 International Symposium on Collaborative Technologies and Systems, Las Vegas, May 14-18, 2006, IEEE Computer Society. IEEE Computer Society. ISBN: 0-7695-2387-0, pp. 38-47.

Demchenko, Y., L. Gommans, C. de Laat, **"Using VO concept for managing dynamic security associations", in "Security and Privacy in Dynamic Environments",** proceedings of the IFIP TC-11 21st International Information Security Conference (SEC2006), 22-24 May 2006, Karlstad, Sweden. Springer. ISBN: 10: 0-387-33405-X, ISBN: 13: 9780-387-33405-X, pp. 377-388.

Yuri Demchenko, Leon Gommans, Cees de Laat, Bas Oudenaarde, **"Web Services and Grid Security Vulnerabilities and Threats Analysis and Model",** Proceedings of the "6th IEEE/ACM International Workshop on Grid Computing", November 13-14, 2005. Seattle, Washington, USA. - pp. 262-267. IEEE Cat. No. 05EX1210C, ISBN 0-7803-9493-3.

Demchenko, Y., L. Gommans, C. de Laat, B.Oudenaarde, A. Tokmakoff, M. Snijders, **"Job-centric Security model for Open Collaborative Environment",** Proceedings 2005 International Symposium on Collaborative Technologies and Systems (CTS2005), May 15-19, 2005, Saint Louis, USA, IEEE Computer Society, ISBN: 0-7695-2387-0, Page 69-77.

Yuri Demchenko, Leon Gommans, Cees de Laat, Bas Oudenaarde, Andrew Tokmakoff, Martin Snijders, Rene van Buuren, **"Security Architecture for Open Collaborative Environment",** Advances in Grid Computing - EGC 2005: European Grid Conference, Amsterdam, The Netherlands, February 14-16, 2005, Revised Selected Papers, LNCS, Springer Verlag, Volume 3470, page 589, (2005).

C.T.A.M. de Laat, H. Blom, L. Gommans, M. Korten, W. Lourens, E.A. van der Meer and B.U. Nideröst, **"The Significance of the New Internet Standards for Collaboratories",** Proc. of the RT'99 Conf., Sante Fé, New Mexico, USA, 9 June 1999. p.468.

8.7 Work relating to our research

A survey on Google scholar showed that several of the lead author publications used for chapters 3 and 4 were are being referred. Below an overview of the most important publications that are being referred.

Authorization of a QoS Path based on Generic AAA was cited by:

Tom DeFanti, Cees de Laat, Joe Mambretti, Kees Neggers, and Bill St. Arnaud. 2003. **TransLight: a global-scale LambdaGrid for e-science.** Commun. ACM 46, 11 (November 2003), 34-41. DOI=10.1145/948383.948407 http://doi.acm.org/10.1145/948383.948407.

Sumit Naiksatam, Silvia Figueira, **Elastic reservations for efficient bandwidth utilization in LambdaGrids,** Future Generation Computer Systems, Volume 23, Issue 1, 1 January 2007, Pages 1-22, ISSN 0167-739X, http://dx.doi.org/10.1016/j.future.2006.02.013.

Mei Lin, Zhangxi Lin, **A cost-effective critical path approach for service priority selections in grid computing economy,** Decision Support Systems, Volume 42, Issue 3, December 2006, Pages 1628-1640, ISSN 0167-9236, http://dx.doi.org/10.1016/j.dss.2006.02.010.

Naiksatam, S.; Figueira, S.; Chiappari, S.A.; Bhatnagar, N., **"Analyzing the advance reservation of lightpaths in lambda-grids,"** Cluster Computing and the Grid, 2005. CCGrid 2005. IEEE International Symposium on , vol.2, no., pp.985,992 Vol. 2, 9-12 May 2005
doi: 10.1109/CCGRID.2005.1558668

He, E.; Xi Wang; Leigh, J., **"A Flexible Advance Reservation Model for Multi-Domain WDM Optical Networks,"** Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference on , vol., no., pp.1,10, 1-5 Oct. 2006.
doi: 10.1109/BROADNETS.2006.4374310

He, E.; Xi Wang; Vishwanath, V.; Leigh, J., **"CAM03-6: AR-PIN/PDC: Flexible Advance Reservation of Intradomain and Interdomain Lightpaths,"** Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE , vol., no., pp.1,6, Nov. 27 2006-Dec. 1 2006.

Lin, Zhangxi, Huimin Zhao, and Sathya Ramanathan. **"Pricing web services for optimizing resource allocation an implementation scheme."** Proc. of the Web2003, Seattle, WA (2003).

Oliver Yu, Anfei Li, Yuan Cao, Leping Yin, Ming Liao, Huan Xu, **Multi-domain Lambda Grid data portal for collaborative Grid applications,** Future Generation Computer Systems, Volume 22, Issue 8, October 2006, Pages 993-1003, ISSN 0167-739X, http://dx.doi.org/10.1016/j.future.2006.03.016.

Lin, Zhangxi, Sathya Ramanathan, and Huimin Zhao. **"Usage-based dynamic pricing of Web services for optimizing resource allocation."** Information Systems and E-Business Management 3.3 (2005): 221-242.

Nut Taesombut, Xinran (Ryan) Wu, Andrew A. Chien, Atul Nayak, Bridget Smith, Debi Kilb, Thomas Im, Dane Samilo, Graham Kent, John Orcutt, **Collaborative data visualization for Earth Sciences with the OptIPuter,** Future Generation Computer Systems, Volume 22, Issue 8, October 2006, Pages 955-963, ISSN 0167-739X, http://dx.doi.org/10.1016/j.future.2006.03.023.

Bobyshev, A., et al. **"A collaborative network middleware project by Lambda Station, TeraPaths, and Phoebus."** Journal of Physics: Conference Series. Vol. 219. No. 6. IOP Publishing, 2010.

Joe Mambretti, Mathieu Lemay, Scott Campbell, Hervé Guy, Thomas Tam, Eric Bernier, Bobby Ho, Michel Savoie, Cees de Laat, Ronald van der Pol, Jim Chen, Fei Yeh, Sergi Figuerola, Pau Minoves, Dimitra Simeonidou, Eduard Escalona, Norberto Amaya Gonzalez, Admela Jukan, Wolfgang Bziuk, Dongkyun Kim, KwangJong Cho, Hui-Lan Lee, Te-Lung Liu, **High Performance Digital Media Network (HPDMnet): An advanced international research initiative and global experimental testbed, Future Generation Computer Systems,** Volume 27, Issue 7, July 2011, Pages 893-905, ISSN 0167-739X, http://dx.doi.org/10.1016/j.future.2010.12.012.

Token-based authorization of Connection Oriented Network resources was (amongst others) cited by:

Beckman, Pete, et al. **"SPRUCE: A system for supporting urgent high-performance computing."** Grid-Based Problem Solving Environments. Springer US, 2007. 295-311.

Chin P. Guok, David Robertson, Evangelos Chaniotakis, Mary Thompson, William Johnston, Brian Tierney. **"A User Driven Dynamic Circuit Network Implementation"**. Lawrence Berkeley National Laboratory. (2009). Retrieved from: http://escholarship.org/uc/item/9pv0k61r

Beckman, P. H., Beschastnikh, I., Nadella, S., & Trebon, N. Beckman, Peter H., et al. **"Building an Infrastructure for Urgent Computing."** High Performance Computing Workshop pp. 75-95, 2006.

Nicolas Trebon, **"Enabling urgent computing within the existing distributed computing infrastructure"**, Dissertation, University of Chicago, 2011, Publication Number, 3472964, http://gradworks.umi.com/3472964.pdf

Franco Travostino, Hoang Doan, **"Grid network middleware."** Ch 7, p113-139, Grid Networks, ISBN-10: 0-470-01748-1, John Wiley & Sons, 2006.

Applications Drive Secure Lightpath Creation across Heterogeneous Domains was (amongst others) cited by:

Van der Ham, J.; Grosso, P.; van der Pol, R.; Toonk, A.; De Laat, C., **"Using the Network Description Language in Optical Networks,"** Integrated Network Management, 2007. IM '07. 10th IFIP/IEEE International Symposium on , vol., no., pp.199,205, May 21 2007-Yearly 25 2007. doi: 10.1109/INM.2007.374784

Guok, C.; Robertson, D.; Thompson, M.; Lee, J.; Tierney, B.; Johnston, W., **"Intra and Interdomain Circuit Provisioning Using the OSCARS Reservation System,"** Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference on , vol., no., pp.1,8, 1-5 Oct. 2006.
doi: 10.1109/BROADNETS.2006.4374316

Chamania, M.; Jukan, A., **"A survey of inter-domain peering and provisioning solutions for the next generation optical networks,"** Communications Surveys & Tutorials, IEEE , vol.11, no.1, pp.33,51, First Quarter 2009. doi: 10.1109/SURV.2009.090104

Ghani, Nasir, Min Peng, and Ammar Rayes. **"Provisioning and survivability in multi-domain optical networks."** WDM Systems and Networks. Springer New York, 2012. 481-519.

Hulsebosch, R. J.; Bargh, M.S.; Fennema, P. H.; Zandbelt, J. F.; Snijders, M.; Eertink, E. H., **"Using Identity Management and Secure DNS for Effective and Trusted User Controlled Light-Path Establishment,"** Networking and Services, 2006. ICNS '06. International conference on , vol., no., pp.79,79, 16-18 July 2006
doi: 10.1109/ICNS.2006.115

Polito, S.G.; Chamania, M.; Jukan, A., **"Extending the Inter-Domain PCE Framework for Authentication and Authorization in GMPLS Networks,"** Communications, 2009. ICC '09. IEEE International Conference on , vol., no., pp.1,6, 14-18 June 2009
doi: 10.1109/ICC.2009.5199021

Chen, X., Zhang, J., Jia, P., Wang, L., Cheng, Y., Zhang, H., & Gu, W.. **"WS-SP: a framework for multi-service provisioning in the next generation optical network"**. In Asia-Pacific Optical Communications (pp. 67841L-67841L). International Society for Optics and Photonics, Nov. 2007.

Chen, Y., Zhang, J., Han, D., Chen, X., Zhao, Y., Gu, W., & Ji, Y.. **"PCE-based service level agreement constraint routing strategy in multi-domain optical network."** In Asia Communications and Photonics (pp. 76331W-76331W). International Society for Optics and Photonics, Nov. 2009.

Adam, G., Bouras, C., Kalligeros, I., Stamos, K., & Zaoudis, I.. **"Security Aspects for Large Scale Distributed Environments".** In SECURWARE 2012, The Sixth International Conference on Emerging Security Information, Systems and Technologies (pp. 7-13) Aug. 2012.

Karmous-Edwards, G., Polito, S. G., Jukan, A., & Rouskas, G. **"A new framework for GLIF Interdomain Resource Reservation Architecture (GIRRA)".** annals of telecommunications-annales des télécommunications, 65(11-12), 723-737, 2010.

Wu, R., & Ji, Y. **"Application-driven grid node architecture for intensive data services on grid based ASON."** In Asia Pacific Optical Communications (pp. 71372F-71372F). International Society for Optics and Photonics, Nov. 2008.

Mambretti, Joe, and Franco Travostino. **"Grid Network Requirements and Architecture."** Grid Networks (2006): 49.

Banaie, Fatemeh, Mohammad Hossein Yaghmaee, and Nazbanoo Farzaneh. **"A blocking probability reduction method in path computation schemes for inter domain networks."** Telecommunications (IST), 2012 Sixth International Symposium on. IEEE, 2012.

Multi-Domain Lightpath Authorization using Tokens is being referred by:

Polito, Silvana Greco, et al. **"Inter-domain path provisioning with security features: Architecture and signaling performance."** Network and Service Management, IEEE Transactions on 8.3 (2011): 219-233.

Paletta, Mauricio, and Pilar Herrero. **"A token-based mutual exclusion approach to improve collaboration in distributed environments."** Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems. Springer Berlin Heidelberg, 2009. 118-127.

Lar, S-U., Xiaofeng Liao, and Syed Ali Abbas. **"Cloud computing privacy & security global issues, challenges, & mechanisms."** Communications and Networking in China (CHINACOM), 2011 6th International ICST Conference on. IEEE, 2011.

Makkes, Marc X., et al. **"Defining intercloud federation framework for multi-provider cloud services integration."** CLOUD COMPUTING 2013, The Fourth International Conference on Cloud Computing, GRIDs, and Virtualization. 2013.

Alfred Wan, Paola Grosso, and Cees de Laat. **"Interoperability of lightpath provisioning systems in a multi-domain testbed."** Testbeds and Research Infrastructures. Development of Networks and Communities. Springer Berlin Heidelberg, 2011. 412-427.

Note: A nice overview, referring to the original authorization concepts that are discussed in this thesis, is given by: Bob Hulsebosch, Jaap Reitsma, Maarten Wegdam, **Federated and Group Management in e-Science.** GigaPort3 deliverable EDS-11R, 2010, http://www.surfnet.hosting.onehippo.com/binaries/content/assets/surf/en/knowledgebase/2011/EDS-11R+Authorizations+and+Group+Management.pdf

8.8 Work relating to our IETF research

After having published our work as IETF documents, we saw OASIS pick up work on the Secure Assertion Mark-up Language [SAML] and referring our work. SAML was considering fulfilling requirements [SAMR] towards a data format for authorization attributes. SAML used authorization framework terminology such as "pushing" and "pulling" data assertions. It also builds on the RAP [RAPWG] terminology (PDP/PEP). SAML was targeted to support multi-domain cases as was described in our work. SAML also recognized that trust negotiations should not be part of its work and should be handled "out-of-band" (i.e. consider it as a business issue).

Examples of publications referring to RFC2903 "Generic AAA Architecture" can be found in the area of:

- History-enabled policy engines [GAMA].
- Solving Key Design Issues for Massively Multiplayer Online Games on Peer-to-Peer Architectures [LUFA].
- Scalable Quality of Service Support for Mobile Users [STAT].
- A single sign-on protocol for distributed web applications based on standard internet mechanisms [GANT].
- Effective Exploitation of Distributed Information for Cooperative Network Security and Routing Optimization [GANT].
- Trust in Machine to Machine communication [ICHA].
- A secure localized authentication and billing (SLAB) scheme [HZHU].
- A AAA survey and a policy-based architecture and framework [RENS].

Examples of publications referring to RFC2904 "AAA Authorization Framework" can be found in the area of:

- Multicast content distribution framework supporting content access control and accounting [HINA].
- Approaches supporting the delegation of privileges [MONT].
- Secure role based messaging, enabling role-oriented secure communication [ZHAO].

- Policy-based access control using a QoS aware network management platform [BERG].
- An opposing definition of authorization in the context of identity management and trusted interaction in Internet and mobile computing [JOSA].
- Consumer Side Resource Accounting in Cloud Computing [MIHO].

9 References

8021	The IEEE 802.1p standard has been merged with the IEEE 802.1d standard for MAC bridges. See http://standards.ieee.org/about/get/802/802.1.html
AAAARG	See charter of concluded AAA Architecture Research group http://irtf.org/concluded/aaaarch
AAAPR	www.science.uva.nl/research/air/projects/aaa
AAATK	For the Generic AAA toolkit work see http://sne.science.uva.nl/aaa/ and http://www.science.uva.nl/research/air/projects/aaa/ (old)
AAAWG	See charter of concluded AAA Working Group: http://www.ietf.org/wg/concluded/aaa
AIRBNB	http://www.programmableweb.com/api/airbnb API for airbnb services
AKAM	For an example of a Content Delivery Network see: www.akamai.com
ALFI	R. Alfieri, R. Cecchini, V. Ciaschini, F. Sparato, L. del'Angello, Á. Frohner, K. Lörentey, **"From gridmap-file to VOMS: managing Authorization in a Grid environment",** Future Generation Computer Systems, Vol. 21, Issue 4, April 2005, Pg 549-558.
AMET	Pierre Amet, "Il y a 5000 ans Elamites inventaient l'écriture," Archeologia 12: 16-23
ANDE	Anne Anderson, Hal Lockhart, **"SAML 2.0 profile of XACML",** Committee Draft 01, http://docs.oasis-open.org/xacml/access_control-xacml-2.0-saml_profile-spec-cd-01.pdf, OASIS 16 September 2004.
ARNA	Bill St. Arnaud, Erik-Jan Bos, Inder Monga, "**GLIF Architecture Green Paper**", GLIF, Jan 2013, http://www.glif.is/publications/papers/GLIF-Architecture-Green-Paper-01-2013.pdf
ASTN	ITU-T Recommendations on the ASTN/ASON Control Plane. http://www.itu.int/ITU-T/studygroups/com15/otn/astncontrol. html
ASTRON	Netherlands Institute for Radio Astronomy, http://www.astron.nl
AUTB	GÉANT2 AutoBAHN – general info available via GÉANT2 homepage [Online]. http://www.geant2.net .
BAC1	Reinhard Bachmann, **Trust, Power and Control in Trans-Organisational Relations**, EGOS Studies 2001, 22/2 pg 337-365, 0170-8406/01 0022-0012
BAC3	Reinhard Bachmann, **The Trust Process in Organisations**, chapter 4, Edward Elgar Publishing, 2003, ISBN 1 84376 078 9
BAS3	Basel Committee on Banking Supervision**, Basel III: A global regulatory framework for more resilient banks and banking systems**, Bank for International Settlements, December 2010, ISBN print: 92-9131-859-0
BERG	Van Den Berghe, S.; De Turck, F.; Demeester, P., **"Integrating policy-based access management and adaptive traffic engineering for QoS deployment,"** Policies for Distributed Systems and Networks, 2004. POLICY 2004. Proceedings. Fifth IEEE International Workshop on , vol., no., pp.211,214, 7-9 June 2004
BERI	G. Bruce Beriman, Steven L. Groom, **"How will astronomy archives survive the data tsunami?"**, Communications of the ACM, Volume 54, Issue 12, December 2011, pg 52-56
BERN	Yoram Bernet, **"Networking Quality of Service and Windows Operating Systems"**, ISBN 1-57870-206-2, New Riders Publishing and Microsoft Corporation, 2001.

BOCA	Jean-Pierre Bocquet-Appel, **"When the World's Population Took Off: The Springboard of the Neolithic Demographic Transition"**, Science 29 July 2011: Vol. 333 no. 6042 pp. 560-561DOI:10.1126/science.1208880
BOND	Alcatel's product announcement Paris, Oct 16th 2002 retrieved from Alcatel's public website in may 2014: http://www.home.alcatel.com/vpr/fullarchive.nsf/032ce43d0ec99b73c12572170035aef2/28e1f63534ce48c8c125723000360d59!OpenDocument
BROW	Maxine D. Brown, "**Blueprint for the future of high-performance networking**", Communications of the ACM, Vol. 46, No 11, November 2003, Pg 30-33.
CA82	**Civil Action No. 82-0192, United States of America**, Plaintiff v.s.Western Electric Company, Inc. and American Telephone and Telegraph Company, Defendants. Plan of Reorganization, Dec. 16th, 1982. http://www.bellsystemmemorial.com/pdf/82-0192.pdf
CATWG	See charter of concluded CAT Working Group: http://www.ietf.org/wg/concluded/cat
CERN	See: http://lhcopn.web.cern.ch/lhcopn/
CHA1	D.W. Chadwick, O. Otenko, E. Ball, **"The PERMIS X.509 Role Based Privilige Management Infrastructure",** IEEE Internet Computing, Volume 7, Issue 2, Mar/Apr 2003, Pg 62-69, http://dx.doi.org/10.1109/MIC.2003.1189190
CHA2	D. Chadwick, S. Otenko, "**A comparison of the Akenti and Permis Authorization Infrastructures"**, Proceedings of Ensuring Security in IT Infrastructures : ITI First International Conference on Information & Communication Technology (ICICT2003), pg 14-34, 30 November – 2 December, 2003, Cairo, Egypt. Edited by Mahmoud T. El-Hadidi
CIAS	Vincenzo Ciaschini, Valerio Venturi, Andrea Ceccanti, "**The VOMS Attribute Certificate Format"**, GFD-I.182, Aug. 2011.
CIM	See: http://www.dmtf.org/standards/cim
CINE	CineGrid Project [Online]. Available at: http://www.cinegrid.org/
CLIN	The Clinton Administration's Policy on Critical Infrastructure Protection: Presidential Decision Directive 63, May 22nd 1998. http://csrc.nist.gov/drivers/documents/paper598.pdf
CRIS	Mihai-Lucian Cristea, Leon Gommans, Li Xu, Herbert Bos, **The token based switch: Per-packet access authorisation to optical shortcuts**, in: Proceedings of IFIP Networking'07, May 2007.
CRIS9	M. Cristea, R.J. Strijkers, D. Marchal, L. Gommans, C. de Laat, R.J. Meijer, "**Supporting Communities in Programmable Networks: gTBN**", IFIP Integrated Management 2009, Colombia University, New York
DAGR	http://eu-datagrid.web.cern.ch/eu-datagrid
DAMI	N. Damianou, N. Dulay, E. Lupu, M Sloman, "**The Ponder Specification Language",** Proc. Policy 2001: Workshop on Policies for Distributed Systems and Networks, Bristol, UK, 29-31 Jan. 2001, Springer-Verlag LNCS 1995, pp. 18-39
DATA	http://datatag.web.cern.ch/datatag
DCN	See: http://www.internet2.edu on the subject DCN / AL2S
DCN	The Internet2 Dynamic Circuit Network [Online]. Available at: http://www.internet2.edu/dcresearch/index.html

DEF1	Thomas A. Defanti, Maxine D. Brown, Cees de Laat**, "Editorial: iGrid2002, The International Virtual Laboratory"**, Future Generation Computing Systems, Volume 19, Issue 6, August 2003, Pg 803-804.
DEFA	Tom DeFanti, Cees de Laat, Joe Mambretti, Kees Neggers, Bill St. Arnaud, "**TransLight: a global-scale LambdaGrid for e-science"**, Communications of the ACM, Volume 46, Issue 11, Nov. 2003, pp. 34-41.
DEM1	Yuri Demchenko, Leon Gommans, Cees de Laat, Bas Oudenaarde, Andrew Tokmakoff, Martin Snijders, Rene van Buuren. ``**Security Architecture for Open Collaborative Environment"** ,"Lecture Notes in Computer Science", Springer-Verlag GmbH, ISSN: 0302-9743, Volume 3470 / 2005. Title: Advances in Grid Computing - EGC 2005: European Grid Conference, Amsterdam, The Netherlands, February 14-16, 2005, http://dx.doi.org/10.1007/11508380_60.
DEM2	Yuri Demchenko, Leon Gommans, Cees de Laat, Bas Oudenaarde, Andrew Tokmakoff, Martin Snijders. **"Job-centric Security model for Open Collaborative Environment"**, - Proceedings 2005 International Simposium on Collaborative Technologies and Systems (CTS2005). - May 15-19, 2005, Saint Louis, USA. - IEEE Computer Society, ISBN: 0-7695-2387-0. - Pp. 69-77.
DEM3	Yuri Demchenko, Olle Mulmo, Leon Gommans, Cees de Laat, Alfred Wan, **"Dynamic security context management in Grid-based applications"**, Future Generations Computer Systems vol. 24 (2008) pg 434-441.
DEM7	Y. Demchenko, L. Gommans, C. de Laat, **Using SAML and XACML for complex resource provisioning in grid based applications**, in: Proceedings IEEE Workshop on Policies for Distributed Systems and Networks, POLICY 2007, Bologna, Italy, 13–15 June 2007 pp. 183–187. ISBN-13: 978-0-7695-2767-3, ISBN-10: 0-7695-2767-1.
DEM8	Y. Demchenko, F. Wan, M. Cristea, C. de Laat, **Authorisation infrastructure for on-demand network resource provisioning** in: Proc. 9th IEEE/ACM International Conference on Grid Computing Grid2008, Sept 29 - Oct 1, 2008, Tsukuba, Japan, page 95-103
DIJK	Freek Dijkstra, Bas van Oudenaarde, Bert Andree, Leon Gommans, Jeroen van der Ham, Karst Koymans, Cees de Laat , **"Control Models at Interconnection Points",** in submission at that time and ultimately published as: **"A Terminology for Control Models at Optical Exchanges",** LCNS, Volume 4543, july 2007, Page 49-60
DIJKG	Freek Dijkstra, Cees de Laat, **Optical Exchanges**, GLIF: http://www.glif.is/publications/papers/20041029KdL_OpticalExchanges.pdf
DLA3	Cees de Laat, Erik Radius, Steven Wallace, **"The Rationale of the Current Optical Networking Initiatives",** iGrid2002 special issue, Future Generation Computer Systems, volume 19 issue 6 (2003).
DRAC	DRAC was adopted by the GLIF community: http://www.glif.is/meetings/2009/tech/peeters-drac.pdf
DRAG	DRAGON Project [Online]. Available at: http://dragon.maxgigapop.net/twiki/bin/view/DRAGON/WebHome

EDUC	Eduroam compliance statement, https://www.eduroam.org/downloads/docs/eduroam_Compliance_Statement_v1_0.pdf[32] http://www.geant.net/Services/UserAccessAndApplications/Pages/eduroam.aspx
EDUR	See: https://www.eduroam.org
EGI	http://www.egi.eu
EIRG	See e-Infrastructure Reflection Group http://www.e-irg.eu
EVLBI	The e-VLBI technique enables real-time data transfer from remote radio telescopes to the central processing facility via optical fibre cables, see: http://www.evlbi.org/
EMI	See: http://www.eu-emi.eu
EMIS	See: http://www.eu-emi.eu/standardization
ESN	http://www.es.net/network/
EXPE	Expedia http://www.programmableweb.com/api/expedia API to book travel.
FERR	D.F. Ferraiolo and D.R. Kuhn, **"Role Based Access Control"** 15th National Computer Security Conference, Baltimore, October 1992.
FIPS	See http://csrc.nist.gov/publications/PubsFIPS.html
FIRG	See: http://redmine.ogf.org/projects/fi-rg/
FLIU	Fang Liu, Jin Tong, Jian Mao, Robert, Bohn, John Messina, Lee Badger and Dawn Leaf, NIST Cloud Computing Reference Architecture, SP500-292, NIST September 2011, http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909505
FOST	Ian Foster, Carl Kesselman, Steve Tuecke, **The Anatomy of the Grid,** The International Journal of High Performance Computing Applications, Fall 2001 vol. 15 no. 3 200-222, Sage Journals, doi: 10.1177/109434200101500302
FRS	**PROHIBITION ON FUNDING OF UNLAWFUL INTERNET GAMBLING**, **Adoption of Unlawful Internet Gambling Enforcement Act**, FEDERAL RESERVE SYSTEM, 12 CFR Part 233 Regulation GG; Docket No. R-1298 - DEPARTMENT OF THE TREASURY 31 CFR Part 132RIN 1505-AB78
GAMA	Gama, Pedro; Ribeiro, C.; Ferreira, P., **"A scalable history-based policy engine,"** Policies for Distributed Systems and Networks, 2006. Policy 2006. Seventh IEEE International Workshop on , vol., no., pp.10 pp.,112, 5-7 June 2006 doi: 10.1109/POLICY.2006.8
GANT	Gantner, Julian, Geyer-Schulz, Andreas, Thede, Anke, **"A single sign-on protocol for distributed web applications based on standard internet mechanisms",** e-Business and Telecommunication Networks, pg 176-183, ISBN 978-1-4020-4760-2, Springer Netherlands, 2006, http://dx.doi.org/10.1007/1-4020-4761-4_14
GART	http://www.gartner.com/it-glossary/hybrid-cloud-computing/
GATED	http://www.merit.edu/research/gated.php
GEA	http://www.geant2.net/server/show/ConWebDoc.2544
GEAS	http://www.geant.net/Services/UserAccessAndApplications/Pages/eduroam.aspx
GEASC	GEANT. (n.d.). GEANT exhibiting at Supercomputing 2012. Retrieved Nov 30, 2012, from www.geant.net: http://www.geant.net/Media_Centre/News/Pages/GEANT_at_SC12.aspx
GENI	See: http://www.geni.net

GFD142	Thijs Metch, Leon Gommans, Egon Grunter, Ralph Niedenberger, Alan de Smet, Gian Luca Volpato, **"Requirements on operating Grids in Firewalled Environments",** OGF 2008, http://www.ogf.org/documents/GFD.142.pdf
GFD173	Guy Roberts, Tomohiro Kudoh, Inder Monga, Jerry Sobieski, John Vollbrecht, **GFD.173 Network Service Interface Framework V1.0**, OGF 2010. http://www.ogf.org/documents/GFD.173.pdf
GFD38	M. Lorch, B. Cowles, R. Baker, L. Gommans, P. Madsen, A. McNab, L. Ramakrishnan, K. Sankar, D. Skow, M. Thompson, **"Conceptual Grid Authorization Framework and Classification"**, GFD.38, Open Grid Forum, Nov. 2004.
GFD83	Ralph Niederberger, William Allcock, Leon Gommans, Egon Grunter, Thijs Metch, Inder Monga, Gian Luca Volpato, Christian Grimm, **GFD-I.083 firewall issues overview**, Open Grid Forum, August 2006.
GHPN	See: http://redmine.ogf.org/projects/ghpn-rg/
GIGA	Zie voor Gigaport project de SURFnet innovatie pagina (2014) http://www.surf.nl/kennis-en-innovatie/innovatieprojecten/surf-onderdeel/surfnet
GLAM	G-lambda Project [Online]. http://www.glambda.net .
GLIF	See: http://www.glif.is
GLIFA	See http://www.glif.is in particular http://www.glif.is/apps
GLMA	GLIF, G. L. (2011). GLIF Maps. Retrieved from http://www.glif.is/publications/maps/
GLOB	See Globus Toolkit security documentation: http://toolkit.globus.org/toolkit/docs/4.0/security/key-index.html
GOLE	Glif Open Lightpath Exchange: http://www.delaat.net/sc/sc10/GLIFAutomatedGOLEPilot.SC.pdf
GOM3	Leon Gommans, Cees de Laat, Bas van Oudenaarde, Arie Taal, "**Authorization of a QoS Path based on Generic AAA**", iGrid2002 special issue, Future Generation Computer Systems, Volume 19, Issue 6 (2003).
GOM4	Leon Gommans, Franco Travostino, John Vollbrecht, Cees de Laat, Robert Meijer, **"Token-based authorization of Connection Oriented Network resources"**, GRIDNETS conference proceedings, oct 2004
GOM5	Leon Gommans, Cees de Laat, Robert Meijer**, Token based path authorization at interconnection points between hybrid networks and a lambda grid**, in: IEEE GRIDNETS2005 proceedings, ISBN 0-7803-9277-9.
GOM61	Leon Gommans, Freek Dijkstra, Cees de Laat, Arie Taal, Alfred Wan, Bas van Oudenaarde, Tal Lavian, Inder Monga, Franco Travostino, "**Applications Drive Secure Lightpath Creation across Heterogeneous Domains"**, IEEE Communication Magazine vol. 44, no 3, March2006.
GOM62	L. Gommans, B. van Oudenaarde, A. Wan, C.T.A.M. de Laat, R. Meijer, F. Travostino, I. Monga, **Token based networking: Experiment NL101, in: iGrid2005**, Future Generation Computer Systems 22 (8) (2006) 1025–1031 (special issue).
GOM8	Leon Gommans, Li Xu, Fred Wan, Yuri Demchenko, Mihai Cristea, Robert Meijer, Cees de Laat , **"Multi-Domain Lightpath Authorization using Tokens",** Future Generation Computing Systems, Vol 25, issue 2, 2008, pp 153-160, DOI 10.1016/j.future.2008.07.013

GOM14	Leon Gommans, John Vollbrecht, Betty Gommans-de Bruijn, Cees de Laat, **"The Service Provider Group Framework**", Future Generation Computing Systems, DOI: 10.1016/j.future.2014.06.002
GRUB	R. Gruber, V. Keller, M. Thiemard, O. Wäldrich, Ph. Wieder, W. Ziegler, P. Manneback, **Integration of grid cost model into ISS/VIOLA meta-scheduler environment**, in: Proceedings of 2nd UNICORE Summit 2006 in Conjunction with EuroPar 2006, Dresden, Germany, in: LNCS, vol. 4375, 2006, pp. 215–224.
GTAF	Globus Toolkit Authorisation Framework [Online]. Available: http://www.globus.org/toolkit/docs/development/4.1.0/security/authzframe/
GUIS	Guisepi, R.A. ed. **"The Origins of Civilizations: The Agrarian Revolution and the Birth of Civilization. Neolithic Revolution."** International World History Project. 2006. World History Center. Retrieved Feb 2014 from http://history-world.org/neolithic.htm
GUOK	Chin P. Guok, David W. Robertson, Evangelos Chaniotakis, Mary R. Thompson, William Johnston, Brian Tierney, **A User Driven Dynamic Circuit Network Implementatio**n, Lawrence Berkeley National Laboratory, 2009, http://escholarship.org/uc/item/9pv0k61r
HEY	Tony Hey, Stewart Tansley, Kristin Tolly, "**The Fourth Paradigm, Microsoft Research",** 2009, ISBN 978-0-9825442-0-4
HINA	Hinard, Y., Bettahar, H., Challal, Y., Bouabdallah, A. **"AAA based security architecture for multicast content distribution",** Proceedings of ISCN'06 7th International Symposium on Computer Networks, Instanbul, (IEEE Cat. No. 06EX1429)
HOUR	Francis Hours, "**Atlas des sites du proche orient (14000 - 5700 BP)"** ISBN 978-2-903264-53-6. Maison de l'Orient méditerranéen, 1994.
HZHU	Haojin Zhu; Xiaodong Lin; Rongxing Lu; Pin-Han Ho; Xuemin Shen, "**SLAB: A secure localized authentication and billing scheme for wireless mesh networks,"** Wireless Communications, IEEE Transactions on , vol.7, no.10, pp.3858,3868, October 2008 doi: 10.1109/T-WC.2008.07418
I1471	See http://www.iso-architecture.org/ieee-1471/defining-architecture.html
ICHA	Inhyok Cha; Shah, Y.; Schmidt, A.U.; Leicher, A.; Meyerstein, M.V., **"Trust in M2M communication,"** Vehicular Technology Magazine, IEEE , vol.4, no.3, pp.69,75, Sept. 2009 doi: 10.1109/MVT.2009.93347
IETF	See http://www.ietf.org
IETF45	Leon Gommans, John Vollbrecht**," Examples of Bandwith Broker Type environments described in Authorization Architecture Concepts"**, Presentation at AAA Working Group Meeting, IETF 45 Oslo Jul. 1999, http://www.ietf.org/proceedings/45/slides/aaa-bandwidth-99jul/sld021.htm
IGRI	See: http://www.igrid2005.org
IN2I	Internet2, Internet2 Innovation Platform FAQ, 2012, http://www.internet2.edu/pubs/Internet2-Innovation-Platform-FAQ.pdf
INCI	INCITS 359-2004 - **Information Technology - Role Based Access Control**, standard published 02/03/2004 by InterNational Committee for Information Technology Standards (formerly NCITS) http://www.techstreet.com/products/1151353
INET2	http://www.internet2.edu

ION	http://www.internet2.edu/ion
IPSWG	See charter of concluded IPSec Working Group: http://www.ietf.org/wg/concluded/ipsec
IRTF	See: http://www.irtf.org
ISO27	See http://www.27000.org/
ISOC	Internet Society, http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet
ISWG	See charter of concluded ISSLL Working Group: http://www.ietf.org/wg/concluded/issll
ITP	Interpay as payment processor, became Equens: http://www.equens.com/aboutus/organisation/history.jsp
JOHN	W. Johnston, S. Mudumbai, M. Thompson, "**Authorization and Attribute Certificates for Widely Distributed Access Control"**, IEEE 7th International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises – WETICE '98. http://escholarship.org/uc/item/5m168628
JONE	J.D. Jones, L. Ong, M.A. Lazer, "**Creating an intelligent optical network worldwide interoperability demonstration**", IEEE Communications Magazine, Volume 42, Issue 11, Nov. 2004.
JOSA	Audun Jøsang. "**Identity Management and Trusted Interaction in Internet and Mobile Computing** (to appear)". IET Information Security http://folk.uio.no/josang/papers/mobtrustid_IET-IFS-Josang.pdf
JWU	Jing Wu, Michel Savoie, Scott Campbell, Hanxi Zhang, Gregor V. Bochmann, Bill St. Arnaud, **Customer-managed end-to-end lightpath provisioning**, International Journal of Network Management 15 (5) (2005) 349–362.
KUDO	Tomohiro Kudoh, Guy Roberts, Inder Monga, Network Services Interface: **An Interface for Requesting Dynamic Inter-datacenter Networks**, NSI paper at OFC2013, http://www.opticsinfobase.org/abstract.cfm?URI=OFC-2013-OM2D.3
LAM	Marice Lambert, "Pourquoi l'écriture est née en Mésopotamie", Archeologia 12: 24-31.
LAMB	http://www.g-lambda.net
LEHM	Tom Lehman, Jerry Sobieski, Bijan Jabbari, DRAGON: **A framework for service provisioning in heterogenous grid network**s, IEEE Communications Magazine 44 (3) (2006).
LG	The authors homepage: http://www.science.uva.nl/~lgommans
LHCG	See The LHC Computing Grid [Online]. http://lcg.web.cern.ch/LCG .
LIV	Livingston Enterprises, role in the development of RADIUS see: http://en.wikipedia.org/wiki/RADIUS
LOBO	J. Lobo, et al., **A policy description language**, in: Proceedings of the American Association for Artificial Intelligence, July 1999.
LUFA	Lu Fan, **"Solving Key Design Issues for Massively Multiplayer Online Games on Peer-to-Peer Architectures"**, PhD thesis, Heriot-Watt University in the School of Mathematical and Computer Sciences May 2009, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.151.2103&rep=rep1&type=pdf
MCDY	McDysan, David E. and Spohn, Darrel L., "**ATM : Theory and Application"**, ISBN 0-07-060362-6, McGraw-Hill series on computer communications, 1995.

MC	See MasterCard annual report 2013 at www.mastercard.com
MCRU	MasterCard Rules July 2011 (versions change frequently) http://www.mastercard.com/us/merchant/pdf/BM-Entire_Manual_public.pdf
MEIJ	Robert J. Meijer, Rudolf J. Strijkers, Leon Gommans, Cees de Laat, **"User Programmable Virtualized Networks"** proceedings of The 2nd IEEE International Conference on e-Science and Grid Computing, Amsterdam, Dec 2006, ISBN: 0-7695-2734-5, page 43
MERIT	Michigan Educational Research Information Triad: http://www.merit.edu
MIHO	Ahmed M Mihoob, **"Consumer Side Resource Accounting in Cloud Computing, PhD Thesis",** University of Newcastle, 2012, https://theses.ncl.ac.uk/dspace/bitstream/10443/1681/1/Mihoob%2012.pdf
MIPWG	See charter of concluded MobileIP Working Group: http://www.ietf.org/wg/concluded/mobileip
MONT	Jose A. Montenegro, Fernando Moya, **"A practical approach of X.509 Attribute Certificate Framework as Support to Obtain Privilege Delegation",** Public Key Infrastructure Lecture Notes in Computer Science Volume 3093, 2004, pp 160-172, Springer Berlin Heidelberg, http://dx.doi.org/10.1007/978-3-540-25980-0_13
MSQU	***La défense de L'Esprit des lois***, Charles de Montesquieu, Barrillot & Fils, Geneve, 1748
NAIK	S. Naiksatam, S. Figueira, **Elastic reservations for efficient bandwidth utilization in LambdaGrids,** Future Generation Computer Systems 23 (1) (2007) 1–22.
NASWG	See charter of concluded NASREQ Working Group: http://www.ietf.org/wg/concluded/nasreq
NCSRA	National Cyber Security Research Agenda -II: https://www.iipvv.nl/sites/stw.demo.infi.nl/files/mediabank/NCSRA-II.pdf
NEGG	Kees Neggers, Position Paper for GEANT High Level Expert Group, Surfnet 18 Januari 2011 www.surfnet.nl/Documents/rapport_201104_SN_Position_Paper_for_GEANT_High_Level_Expert_Group.pdf
NEOW	See: http://en.wikipedia.org/wiki/Neolithic_Revolution
NEXT	See NextGRID: ftp://ftp.cordis.europa.eu/pub/ist/docs/grids/nextgrid_fact_sheet.pdf
NFV	Network Functions Virtualisation: http://www.etsi.org/technologies-clusters/technologies/nfv
NIST	Security of Federal Automated Information Resources, Memorandum for the heads of departments and agencies, June 23rd 1999 http://csrc.nist.gov/drivers/documents/SecFedAIS.pdf
NLIG	Netherlight: http://www.surf.nl/en/services-and-products/netherlight/index.html
NOO1	Bart Nootenboom, Frederique Six, **The Trust process in Organisations**, chapter 1, Edward Elgar Publishing, 2003, ISBN 1 84376 078 9
NOO3	Bart Nootenboom, **The Trust Process in Organisations**, Edward Elgar Publishing, 2003, ISBN 1 84376 078 9
NORT	William B. Norton, "**A business case for ISP Peering"**, www.equinix.com/pdf/whitepapers/Business_case.pdf
NPG	John Vollbrecht, Leon Gommans, **"Operating Framework for a Virtual Connection Network"**, OGF 36 NSI WG Chicago, Oct 10th, 2012.,Presentation: http://redmine.ogf.org/dmsf_files/10197?download= Paper: http://redmine.ogf.org/dmsf_files/10192?download=.

NSFN	For NSFnet and the role of Merit in the history of Internet, see: http://en.wikipedia.org/wiki/History_of_the_Internet
NSI	See: http://redmine.ogf.org/projects/nsi-wg/
OCCI	The Open Cloud Computing Interface working group: http://occi-wg.org
OGF	See: http://www.ogf.org
OGSA	See: http://redmine.ogf.org/projects/ogsa-authz-wg/
OLIV	Francesco Oliviero, **"On the Effective Exploitation of Distributed Information for Cooperative Network Security and Routing Optimization",** PhD Thesis, Università Degli Studi di Napoli Federico II, Facoltà di Ingegneria, Dipartimento di Informatica e Sistemistica, November 2007, http://www.fedoa.unina.it/2063/1/Oliviero_Ingegneria_Informatica_Automatica.pdf
OMB	Circular No A-130 establishing policy for the management of Federal information resources. Office of Management and Budget, Nov. 2000, http://csrc.nist.gov/drivers/documents/a130trans4.pdf
ONF	See: http://www.opennetworking.org
OPPE	A. Leo Oppenheimer, **"On an Operational Device in Mesopotamian Bureaucracy",** Journal of Near Eastern Studies, Vol. 18, No. 2 (Apr., 1959), pg. 121-128. http://www.jstor.org/stable/543273
OPTI	See OptIPuter Project [Online]. http://www.optiputer.net
OSBO	Osborn, S.L., **"Mandatory Access Control and Role-Based Access Control Revisited"**, Proceedings of Second ACM Workshop on Role-Based Access Control, Nov. 1997.
OTWG	See Optical Transport Working Group at: https://www.opennetworking.org/working-groups/optical-transport
OUD5	S.M.C.M. van Oudenaarde, Z.W. Hendrikse, F. Dijkstra, L.H.M. Gommans, C.T.A.M. de Laat, R.J. Meijer, **"An Open Grid Services Architecture Based Prototype for Managing End-to-End Fiber Optic Connections in a Multi-Domain Network"** in High-Speed Networks and Services for Data-Intensive Grids: the DataTAG Project, Special Issue Future Generation Computer Systems, Volume 21, Issue 4 (2005).
OUDE	S. van Oudenaarde, Z. Hendrikse, F. Dijkstra, L. Gommans, C. de Laat, R.J. Meijer, "**Dynamic paths in multi-domain optical networks for grids",** Future Generation Computer Systems 21 (2005) pg 539-548.
PARQ	See: http://parquet.io
PERM	PERMIS Project [Online]. Available: http://sec.cs.kent.ac.uk/permis/
PFDR	M. Stevens, W. Weiss, H. Mahon, B. Moore, J. Strassner, G. Waters, A. Westerinen, J. Wheeler, **"Policy Framework",** IETF Internet draft,1999, http://tools.ietf.org/html/draft-ietf-policy-framework-00.txt
PFLDR	Strassner, J., Ellesson, E., Moore, B., Moats, R.. "**Policy framework ldap core schema"**. IETF Internet draft, 2001, http://www.ietf.org/proceedings/51/I-D/draft-ietf-policy-core-schema-11.txt

PFWG	See the IETF Policy Framework Working Group charter at: http://datatracker.ietf.org/wg/policy/charter/
PHOS	The Phosphorus Project [Online]. Available at: http://www.ist-phosphorus.eu
PIEP	Pieper, G., T.A. DeFanti, Q. Liu, M. Katz, P. Papadopoulos, J. Keefe, G. Hidley, G. Dawe, I. Kaufman, B. Glogowski, K. Doerr, J.P. Schulze, F. Kuester, P. Otto, R. Rao, L. Smarr, J. Leigh, L. Renambot, A. Verlo, L. Long, M. Brown, D. Sandin, V. Vishwanath, R. Kooima, J. Girado, B. Jeong, "**Visualizing Science: The OptIPuter Project**," SciDAC Review, Issue 12, Spring 2009, IOP Publishing in association with Argonne National Laboratory, for the US Department of Energy, Office of Science, pp. 32-41.
PKIXWG	See charter of concluded PKI X.509 Working Group: http://www.ietf.org/wg/concluded/pkix
POLWG	See: http://datatracker.ietf.org/wg/policy/charter/
POST	J.N. Postgate, **"Early Mesopotamia: society and economy at the dawn of history"**, Routledge, London/New York, 1992, ISBN 0-415-00843-3. Online: http://cdli.ucla.edu/staff/englund/m104/m104readings/Postgate_b.pdf
QBON	Internet2 QBone bandwidth broker project: http://qbone.internet2.edu/bb Internet2 QBone: A Test Bed for Differentiated Services: http://www.isoc.org/inet99/proceedings/4f/4f_1.htm
R1172	Perkins D., Hobby R. **"The Point-to-Point Protocol (PPP) Initial Configuration Options"**. RFC 1172, IETF, July 1990.
R1510	J. Kohl, C. Neuman, **"The Kerberos Authentication Service (V5)",** RFC1510, IETF Sept. 1993.
R1633	R. Braden, D. Clark, S. Shenker, **"Integrated Services in the Internet Architecture: an Overview"**, RFC1633, IETF, June 1994.
R1771	Y. Rekhter, et al., **RFC 1771 A Border Gateway Protocol 4 (BGP-4),** IETF March 1995.
R1930	J. Hawkinson, T. Bates, **"Guidelines for creation, selection, and registration of an Autonomous System"**, RFC1930, IETF 1996, http://www.ietf.org/rfc/rfc1930.txt
R1994	Simpson, W., **"PPP Challenge Handshake Authentication Protocol (CHAP)"**, RFC1994, IETF, August 1996.
R2078	J. Linn, **"Generic Security Service Application Programming Interface, Version 2",** RFC2078, IETF Jan. 1997.
R2138	Rigney, C., Rubens, A., Simpson, W. and S. Willens, **"Remote Authentication Dial In User Service (RADIUS)",** RFC 2138, IETF, April 1997
R2205	R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, **"Resource ReSerVation Protocol (RSVP)"**, RFC2205, IETF Sept. 1997.
R2246	T. Dierks, C. Allen, **"The TLS Protocol Version 1.0"**, IETF, Jan. 1999.
R2251	Wahl, M., Howes, T., and S. Kille, **"Lightweight Directory Access Protocol (v3)"**, RFC 2251, IETF Dec. 1997.
R2284	L. Blunk, J. Vollbrecht, **"PPP extensible Authentication Protocol (EAP)",** RFC 2284, IETF Mar 1998.
R2459	Housley, R., Ford, W., Polk, W. and D. Solo, **"Internet X.509 Public Key Infrastructure -- Certificate and CRL Profile"**, RFC 2459, IETF, January 1999.

R2474	Nichols K., Blake S., Baker F., Black D., **"Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers"**, RFC2474, IETF, December 1998
R2607	B. Aboba, J. Vollbrecht, **"Proxy Chaining and Policy Implementation in Roaming"**, RFC 2607, IETF, June 1999.
R2643	D. Ruffen, T. Len, J. Yanacek, **"Cabletron's SecurFast VLAN Operational Model Version 1.8",** IETF, Aug. 1999.
R2704	Blaze, M., Feigenbaum, J., Ioannidis, J. and A. Keromytis, "The KeyNote Trust-Management System Version 2", RFC 2704, September 1999.
R2743	J. Linn, **"Generic Security Service Application Program Interface Version 2, Update 1",** RFC2743, IETF, Jan 2000.
R2748	D. Durham, J. Boyle, R. Cohen, S. Herzog, R. Rajan, A. Sastry, "**The COPS (Common Open Policy Service) Protocol",** RFC 2748, IETF Januari 2000.
R2750	S. Herzog, **"RSVP extensions for Policy Control"**, RFC2750, IETF, Jan 2000.
R2753	Yavatkar, R., Pendarakis, D., and R. Guerin, **"A Framework for Policy-based Admission Control"**, RFC 2753, IETF, Jan. 2000.
R2801	D. Burdett, **"Internet Open Trading Protocol – IOTP"**, IETF, Apr. 2000
R2818	E. Rescorla, **"HTTP over TLS",** RFC2818, IETF, May 2000.
R2865	C. Rigney, S. Willens, A. Rubens, W.Simpson, **"Remote Authentication Dail In User Service (RADIUS)",** RFC2865, IETF June 2000
R2903	C. de Laat, G. Gross, L. Gommans, J. Vollbrecht, D. Spence, "**Generic AAA Architecture",** RFC2903, IETF, Sept. 2000.
R2904	J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, **"AAA Authorization Framework"**, RFC2904, IETF Aug. 2000.
R2904I	See introduction of AAA Authorization Framework [R2904]
R2905	J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, **"AAA Authorization Application Examples"**, RFC2905, IETF, Aug. 2000.
R2906	S. Farrell, J. Vollbrecht, P. Calhoun, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, "**AAA Authorization Requirements"**, RFC2906, IETF, Aug. 2000.
R2998	Y. Bernet, P. Ford, R. Yavatkar, F. Baker, L. Zhang, M. Speer, R. Braden, B. Davie, J. Wroclawski, E. Felstaine, "**A Framework for Integrated Services Operation over Diffserv Networks**", RFC 2998, IETF, Nov. 2000
R3031	Rosen, E., Viswanathan, A., and R. Callon, **"Multiprotocol Label Switching Architecture"**, RFC 3031, IETF, Jan. 2001.
R3060	Moore, B., Ellesson, E., Strassner, J., & Westerinen, A.,. **"Policy core information model– version 1 specification".** RFC3060, IETF, Feb 2001
R3084	**RFC 3084 COPS Usage for Policy Provisioning (COPS-PR**), K. Chan, J. Seligson, D. Durham, S. Gai, K. McCloghrie, S. Herzog, F. Reichmeyer, R. Yavatkar, A. Smith, IETF March 2001.
R3281	Farrell S., Housley R., **"An Internet Attribute Certificate Profile for Authorization"**, RFC3281, IETF, April 2002.

R3313	**RFC 3313 Private Session Initiation Protocol (SIP) Extensions for Media Authorization**, W. Marshall, IETF January 2003.
R3473	Berger, L., Ed., **"Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions"**, RFC 3473, IETF, Jan. 2003.
R3566	S. Frankel, H. Herbert, "**The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec"**, RFC3566, IETF 2003, http://www.ietf.org/rfc/rfc3566.txt
R3588	P. Calhoun, J. Loughney, E. Guttman, G. Zorn, J. Arkko, "**Diameter Base Protocol",** RFC 3588, IETF, Sept 2003.
R5151	A. Farrell, A. Ayyangar, J.P. Vasseur, **Inter-domain MPLS and GMPLS traffic engineering—resource reservation protocol-traffic engineering (RSVPTE) extensions**, RFC 5151, IETF Februari 2008.
R791	J. Postel, **"Internet Protocol"**, RFC791, IETF 1981, http://www.ietf.org/rfc/rfc791.txt
R826	Plummer D., **"An Ethernet Address Resolution Protocol"**, RFC826, IETF November 1982.
RAJY	Raj Yavatkar, **"COPS and RAP Overview",** presentation made during the AAA BoF meeting at IETF42 http://www.ietf.org/proceedings/42/slides/aaa-yavatkar-98aug.pdf
RAMA	Karthik Ramasamy, "**Taking Hadoop to Enterprise Security Standards**", presentation at Hadoop Summit Amsterdam April 2-3, 2014, http://hadoopsummit.org
RAPWG	See the IETF Resource Allocation Protocol Working Group charter at http://datatracker.ietf.org/wg/rap/charter/
RENS	Rensing, C.; Karsten, M.; Stiller, B., **"AAA: a survey and a policy-based architecture and framework,"** Network, IEEE , vol.16, no.6, pp.22,27, Nov/Dec 2002,. doi: 10.1109/MNET.2002.1081762
ROWG	See charter of concluded roamops Working Group: http://www.ietf.org/wg/concluded/roamops
SAML	See https://www.oasis-open.org/standards#samlv1.0
SAMR	"OASIS Security Services Use Cases And Requirements", consensus draft 1, May 2001. https://www.oasis-open.org/committees/security/docs/draft-sstc-saml-reqs-01.pdf
SAN00	R. Sandhu, D. Ferraiolo, R. Kuhn, **"The NIST Model for Role-Based Access Control: Towards a Unified Standard",** In *Proceedings of the fifth ACM workshop on Role-based access control (RBAC '00).* ACM, New York, NY, USA, 47-63. DOI=10.1145/344287.344301 http://doi.acm.org/10.1145/344287.344301
SAN96	R. S. Sandhu, E.J. Coyne, H.L. Feinstein, C.E. Youman, **"Role-Based Access Control Models"**, IEEE Computer 29(2): 38-47, IEEE Press, 1996
SAN98	Sandhu, R., Q. Munawer. **"How to do Discretionary Access Control Using Roles.**" In Proc. of 3rd ACM Workshop on Role Based Access Control (RBAC-98), Fairfax, VA, USA, October 1998, ACM Press.
SARN	SARNET announcement: http://gss.uva.nl/news-and-events/content3/2014/08/nwo-grant-for-internet-security.html
SCHM	Denise Schmandt-Besserat. "**Before Writing, volume 1: from counting to cuneiform**.", University of Texas Press, Austin,1992.

SCOR	See: http://www.supercomputing.org
SDN	See: http://en.wikipedia.org/wiki/Software-defined_networking
SHAM	Adi Shamir, **Identity-based cryptosystems and signature schemes**, in: Advances in Cryptology: Proceedings of CRYPTO 84, in: Lecture Notes in Computer Science, vol. 7, 1984, pp. 47–53.
SHVA	Shvachko, K.; Hairong Kuang; Radia, S.; Chansler, R., **"The Hadoop Distributed File System,"** *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on* , vol., no., pp.1,10, 3-7 May 2010 doi:10.1109/MSST.2010.5496972
SMARR	Smarr, Larry, **"The OptIPuter and Its Applications"** 2009 IEEE LEOS Summer Topicals Meeting on Future Global Networks, July 22, 2009, pp. 151-152, doi: 10.1109/LEOSST.2009.5226201
SNE	See: http://sne.science.uva.nl
SSHWG	See charter of concluded Secsh Working Group: http://www.ietf.org/wg/concluded/secsh
STARL	Starlight: http://www.startap.net/starlight/ABOUT/
STAT	Gűnther Stattenberger, **"Scalable Quality of Service Support for Mobile Users"**, PhD Thesis, Institut für Informatik und angewandte Mathematik, University of Bern, December 2012, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.118.4379&rep=rep1&type=pdf
STRA	John Strassner, **"Directory Enabled Networks",** ISBN 1-57870-140-6, Mcmillan Technical Publishing, 1999.
STRI	Rudolf Strijkers, Willem Toorop, Alain van Hoof, Paola Grosso, Adam Belloum, Dmitry Vasuining, Cees de Laat, Robert Meijer, **"AMOS: Using the Cloud for On-Demand Execution of e-Science Applications,"** in IEEE e-Science 2010 Brisbane, Australia: IEEE Computer Society
SURF	See SURFnet [Online]. Available at: http://www.surfnet.nl .
TAKE	Atsuko Takefusa, et al., G-Lambda: **Coordination of a grid scheduler and lambda path service over GMPLS**, in: iGrid2005, Future Generation Computer Systems 22 (8) (2006) 868–875 (special issue).
TCSE	Trusted Computer Security Evaluation Criteria, DOD 5200.28-STD. Department of Defense, 1985.
TEPO	Terena Mobility Task Force, Deliverable I: TF-Mobility roaming policy document, version 1.2, http://www.terena.org/activities/tf-mobility/deliverables/delI/Roaming_policy_document_v.1.2.pdf, Terena 2003
TFMD	Leon Gommans, John Vollbrecht, **" Trust Framework for Multi-Domain Authorization",** Internet2 Spring Member Meeting, Arlington Virginia, April 25th 2012, http://events.internet2.edu/2012/spring-mm/agenda.cfm?go=session&id=10002327&event=1036
TFMO	http://www.terena.org/activities/tf-mobility/
TIER	Brian Tierney, Ezra Kissel, Martin Swany, Eric Pouyoul, "**Efficient Data Transfer Protocols for Big Data"**, Proceedings of the 2012 IEEE 8th International Conference on E-Science, Chicago, IL, October 2012. IEEE ComSoc ISBN: 987-1-4673-4467-8.
TLSWG	See charter of TLS Working Group: http://datatracker.ietf.org/wg/tls/charter/

TMF	Telecommunication Management Forum Frameworx effort (2014): http://www.tmforum.org/TMForumFrameworx/1911/home.html
TRAD	F. Travostino, Rob Keates, Tal Lavian, Inder Monga, Bruce Schofield, **"Project DRAC: Creating an applications-aware network"**, Nortel Technical Journal, February 2005, pp. 23-26. http://www.cs.berkeley.edu/~tlavian/publications/article/DRAC-Nortel_Journal.pdf
TRAV	F. Travostino, P. Daspit, L. Gommans, C. Jog, C.T.A.M. de Laat, J. Mambretti, I. Monga, B. van Oudenaarde, S. Raghunath, P.Y. Wang, **Seamless live migration of virtual machines over the MAN/WAN**, in: iGrid2005, Future Generation Computer Systems 22 (8) (2006) 901–907 (special issue).
UBER	https://developer.uber.com/ Developer API for UBER services.
UCLP	UCLP [Online]. http://www.canarie.ca/canet4/uclp/ .
VISA	http://en.wikipedia.org/wiki/Visa_Inc
WADA	http://www.wada-ama.org/
WSTL	Web Services Trust Language (WS-Trust) [Online]. Available at: ftp://www6.software.ibm.com/software/developer/library/ws-trust.pdf.
X509	Recommendation X.509 | ISO/IEC 9594-8: **Public-key and attribute certificate frameworks,** ITU-T, http://www.itu.int/rec/T-REC-X.509-201210-I/en
XAML	XACML: eXtensible Access Control Markup Language, OASIS standard [Online]. http://www.oasis-open.org
XEN	www.cl.cam.ac.uk/Research/SRG/netos/xen
ZHAO	Gansen Zhao; Chadwick, D.W., **"Evolving messaging systems for secure role based messaging,"** Engineering of Complex Computer Systems, 2005. ICECCS 2005. Proceedings. 10th IEEE International Conference on , vol., no., pp.216,223, 16-20 June 2005
ZIMA	Paul Zimansky: " **Review of Before Writing,"** by Denise Schmandt-Besserat , Journal of Field Archaeology Vol. 20, No. 4 (Winter, 1993), pp. 513-517, http://www.jstor.org/stable/530080

Summary

In this thesis we show what is needed to build a generic multi-domain authorization system. When placed in e-Infrastructure context, such system is capable of allowing scientific applications to access combinations of infrastructure components. These components are delivered as a chain by multiple service providers. The research emerged from the notion that automatic creation of service chains will need an authorisation system. A multi-domain authorization system allows different service providers to work together when automatically delivering service chains, whilst retaining the ability to define own access policies. Maintaining autonomy amongst Service Providers is an essential requirement. To allow authorization transaction to happen, involved parties must trust each other. To be trusted in a chain, each service provider must know that any policy rule it executes is correct. Such trust emerges from a common set of rules that may need to be enforced depending on the risk involved.

In our research to the question what is needed to build a multi-domain authorization system, we recognized that the question must be approached from at least two viewpoints:

- **The engineering viewpoint,** where different authorization transaction scenario's must be supported by different functions and protocols.
- **The business viewpoint,** where fulfilment of service agreements in mutual trust is key.

Our research was performed in three phases over a period of 15 years. Phase one and two considers the engineering viewpoint. The third phase considers the business viewpoint along with the engineering viewpoint.

The engineering viewpoint.

From the engineering viewpoint we asked ourselves questions like: "What generic authorization functions can be distinguished?", "How do they interact?", "What concepts are expected to work best in multi-domain scenarios?" and "How can these concepts be applied?"

In our research, performed in phase one within the context of the Internet Engineering Task Force, we proposed an Authorization Framework and Generic AAA Architecture to describe and handle authorization transactions. The Framework recognizes a number of fundamental authorization sequence models. The Architecture describes functional elements that can generically handle authorization transactions across multiple domains. Using a number of example scenarios we motivated that our proposal could be generically applicable. We also recognized a number requirements for a design.

Phase two of our research focussed on the question how the Authorization Framework and Generic AAA Architecture concepts can be applied to perform multi-domain authorization.
The inherent research orientation of National Research and Education Networks, and the need for these autonomous organisations to collaborate in order to provide dedicated network

connections at global scale, legitimized our research questions to be placed in this context. We transformed our research questions into more specific questions like "What generic concepts work best for classes of applications that use multi-domain network resources?" and subsequently "How can these concepts be applied to optical networking?". This phase of our research explores and demonstrates the use of two of our Framework sequence models and its combination, using tokens, to authorize the use of multi-domain network segments.

We first hypothesized that the "Agent" model (whereby a request is first send to the authority and subsequently the authority provisions the connection) would be most suitable. Based on experiments with this model we concluded that this model, when applied to multi-domain scenario's, would be potentially too slow to handle requests. By separating the request for a connection from signalling the fact an application wants to use a connection, we concluded that a combination of the agent model with a model whereby the authority issues a token that can be used to access the connection at the desired moment (so called "push" model) is a more suitable alternative. With experiments we have demonstrated that this combined (so called "token") model can be implemented in different ways in network environments. We show that a simple token, that within a domain only refers to a meaning of a token (the meaning of a token can be different for each domain) allows a domain to preserve its autonomy as much as possible. The token can point in each domain to something that must be done correctly. Each domain will determine what the correct thing is that should be done. As such, phase two validates that the functionalities, described by our Generic AAA Architecture, can handle such scenarios.

The business viewpoint.

In phase three of our research we ask the question *"What is needed to arrange trust when authorizing e-infrastructure resources?"* We already saw in phase one that *trust relationships are necessary for authorization transactions to take place*. Based on a study of existing examples from the payment world and the educational roaming world, we created a framework describing the organisation of "Service Provider Groups". This framework makes the often implicit assumed ways explicit of how rules and agreements are transformed in to policies that determine what the *correct things* are that must be done to achieve the desired trust in the operation of a system.

We foresee that our models are in particular applicable to scenario's where chains of electronic services are created automatically and offered as a single service to users. Currently we increasing see the appearance of such chained services that are built using Cloud type services and services that are built via so called Application Programming Interfaces (API's). Joining these autonomous service raises the question who is going to act as party that takes liability for an offered service chain as a whole. Same as in our studied example from the Credit Card world, it takes a parties such as MasterCard to take liability for handling authorization transactions in collaboration with banks as autonomous service providers. When thinking about what is needed to build an authorization system in such context, this research contributes by recognizing the necessary functional elements, both on technical and business side, by using a number of frameworks and a technical architecture that has been validated for its applicability.

Samenvatting

In dit proefschrift laten we zien wat er voor nodig is om een generiek multi-domein autorisatie systeem te bouwen. Wanneer geplaatst binnen de context van e-Infrastructuren, is een dergelijk systeem in staat wetenschappelijke toepassingen toegang te verlenen tot combinaties van infrastructuur componenten. Deze componenten worden door meerdere dienstverleners in een keten geleverd. Het onderzoek is ontstaan vanuit de gedachte dat het automatisch samenstellen van diensten ketens een autorisatie systeem nodig heeft. Een multi-domein autorisatie systeem stelt verschillende dienstverleners in staat om samen te werken bij het automatisch aanbieden van dienstenketens, terwijl iedereen de mogelijkheid behoudt om eigen toegangsregels te stellen. Behoud van autonomie tussen dienstverleners vormt een essentiële eis. Om autorisatie transacties uit te kunnen voeren, zullen de betrokken partijen elkaar moeten vertrouwen. Om in een keten vertrouwd te kunnen worden, moet iedere autonome dienstverlener weten dat iedere uit te voeren beleidsregel (policy) juist is. Een dergelijk vertrouwen ontstaat uit een gezamenlijke set van regels die op basis van risico al dan niet gehandhaafd wordt.

In ons onderzoek naar de vraag wat er voor nodig is om een multi-domein autorisatie systeem te bouwen, onderkenden we dat de vraag in ieder geval vanuit twee gezichtspunten dient te worden benaderd:

- **Het engineering gezichtspunt,** waarbij verschillende autorisatie transactie scenario's met verschillende functies en protocollen ondersteund moeten worden.
- **Het zakelijk gezichtspunt,** waarbij het nakomen van service afspraken in onderling vertrouwen centraal staat.

Het onderzoek omvat werk gedaan in drie fasen over een periode van 15 jaar. Fase een en twee beschouwen het engineering gezichtspunt. De derde fase onderzoekt het zakelijk gezichtspunt naast het engineering gezichtspunt.

Het engineering gezichtspunt.

Vanuit het engineering gezichtspunt stelden we ons vragen zoals "Welke generieke autorisatie functies kunnen we onderscheiden?", "Hoe werken deze functies samen?", "Welke concepten denken we dat het beste werken voor multi-domein scenario's?" en "Hoe toepasbaar zijn deze concepten?"

In ons onderzoek van fase een, gedaan in Internet Engineering Task Force kader, hebben we een Autorisatie Raamwerk en een Generieke AAA Architectuur voorgesteld waarmee autorisatie transacties stromen beschreven c.q. afgehandeld kunnen worden. Het Raamwerk herkent een aantal karakteristieke autorisatie volgorde-modellen. De Architectuur beschrijft functionele elementen waarmee autorisatie transacties over meerdere domeinen heen verwerkt kunnen worden. Aan de hand van een aantal voorbeeld scenario's hebben we gemotiveerd dat ons voorstel algemeen toepasbaar zou moeten zijn. Daarnaast hebben we een aantal ontwerp eisen herkend.

Fase twee van ons onderzoek was gericht op de vraag of ons Autorisatie Raamwerk en de Generieke AAA Architectuur concepten toepasbaar zijn bij het uitvoeren van multi-domein autorisaties.

De inherent onderzoek gerichtheid van Nationale Research en Educatie Netwerken en de noodzaak tot samenwerking van deze autonome organisaties om speciale netwerkverbindingen op wereldwijde schaal te kunnen leveren, legitimeerde het stellen van onze onderzoeksvragen in deze context. De onderzoeksvragen werden derhalve verbijzonderd tot: "Welke generieke concepten werken het beste voor toepassingen die gebruik maken van multi-domein netwerk voorzieningen? en vervolgens "Hoe kunnen deze concepten het beste worden toegepast op optische netwerken?" Deze fase van ons onderzoek bekijkt en demonstreert het gebruik van twee van onze Raamwerk modellen en een combinatie waarbij, met behulp van tokens, multi-domein netwerk segmenten worden geautoriseerd.

We stelden allereerst dat het "Agent" model (waarbij een verzoek eerst naar de autoriteit gestuurd wordt en de autoriteit vervolgens de verbinding tot stand brengt) het meest geschikte model zou zijn. Op basis van experimenten met dit model kwamen we tot de conclusie dat dit model, toegepast in multi-domein omgevingen, potentieel te traag zou zijn met het afhandelen van verzoeken. Door de vraag naar een verbinding te scheiden van het aangeven van het feit dat een applicatie gebruik wil maken van een verbinding, kwamen we tot de conclusie dat een combinatie van het agent model met het model waarbij de autoriteit een token afgeeft dat op het gewenste moment toegang tot de verbinding geeft (het zgn. "push" model), een geschiktere oplossing vormt. Met experimenten hebben we aangetoond dat dit gecombineerde (het zgn. "token") model op verschillende manieren in een netwerk omgeving implementeerbaar is. We laten zien dat een eenvoudig token, dat uitsluitend binnen een domein verwijst naar de bedoeling van het token (de bedoeling kan voor ieder domein immers anders zijn) de autonomie van een domein zoveel mogelijk behouden kan worden. Het token kan binnen een domein verwijzen naar iets wat "juist" gedaan moet worden. Een domein bepaald daarbij zelf wat het "juiste" is. Als zodanig, valideert fase twee dat de functionaliteiten, beschreven door de Generieke AAA Architectuur, dit soort scenario's kunnen afhandelen.

Het zakelijk gezichtspunt.

In fase drie stelden we ons vanuit zakelijk gezichtspunt de vraag: *"Wat is er nodig om vertrouwen te regelen bij het autoriseren van e-Infrastructuur middelen?"* In fase 1 zagen we al dat *vertrouwensbanden noodzakelijk zijn om autorisatie transacties te kunnen laten plaatsvinden.* Op basis van onderzoek naar bestaande voorbeelden uit de betalingswereld en de Educatieve WiFi roaming wereld is een raamwerk bedacht dat de organisatie van dienstverlener groepen beschrijft ("Service Provider Groups"). Dit raamwerk maakt de vaak impliciet aangenomen manieren expliciet hoe regels en afspraken omgezet kunnen worden naar beleidsregels die bepalen wat de *"juiste dingen"* zijn die gedaan moeten worden om het gewenste vertrouwen in de werking van een systeem te kunnen opbouwen.

Onze modellen zien wij vooral toepasbaar in werelden waarbij automatisch ketens van elektronische diensten samengesteld moeten worden die vervolgens aan gebruikers worden aangeboden als een enkele dienst. Heden ten dage zien we bijvoorbeeld steeds meer aaneenschakelingen ontstaan

tussen Cloud diensten en diensten van bedrijven die via zogenaamde Application Programming Interfaces (API's) geleverd worden. Bij aaneenschakelingen van autonome diensten ontstaat de vraag wie als risicodragende partij voor het geheel wil fungeren. Net als bij het in ons onderzoek genoemde Credit Card voorbeeld, is er een partij als MasterCard nodig die als risicodragende partij samen met banken als autonome dienstverleners autorisatie transacties afhandelt.

Als we denken over wat er voor nodig is om in een dergelijke context een autorisatie systeem te bouwen, dan draagt dit onderzoek bij aan het herkennen van de noodzakelijke functionele elementen, zowel aan de engineering als zakelijke kant, door middel van een aantal raamwerken en een functionele architectuur die op haar toepasbaarheid is onderzocht.

Research project acknowledgements.

Apart from being supported by Air France–KLM, the work on this thesis has been supported by the SURFnet GigaPort Research on Networks project and the COMMIT (P20.3) project.

The following research projects formed the context and the basis in terms of funding, facilities and source of knowledge that allowed this research to be performed:

National projects:
SURFnet GigaPort / GigaPort2 / GigaPort NG projects
NWO DAS-2, DAS-3 / Virtual Lab for e-Science (VL-e) projects
Collaboratory.nl
NWO StarPlane
NWO SCARIe
TNO-FEL: Network topologies.
COMMIT-WP20.3 SeSI.

EU Projects:
DataTAG IST 2001-32459
DataGRID IST FP5 (2001)
EGEE-I and EGEE-II projects
NextGrid: IST FP6, contract 511563
Phosphorus IST FP6 contract 034155
GÉANT GN3

International Collaborations
OptiPuter project
Global Lambda Integrated Facility (GLIF) / LambdaGrid project
Internet2 DCN/ION project
ESnet Dragon project
CineGrid project

Corporate Support:
Cabletron Systems
Enterasys Networks
Ellacoya Networks
Cisco
Glimmerglass
Calient
Intel Corporation
Nortel Networks / Ciena
Level3
Alcatel
i-Beleon

Acknowledgements

The completion of this thesis would not have been possible without the help of many people that supported this endeavour. There is a long list of people that contributed by considering, supporting, funding, evaluating, discussing and commenting on the concepts and experiments presented. A few people have however played a special role that I like to acknowledge. I am in particular very grateful to:

Prof. dr. ir. Cees de Laat, who offered me the unique opportunity to research the AAA topic at University of Utrecht and at University of Amsterdam. I foremost thank Cees for believing in me as he allowed me to become part of his research group that allowed me to take this research forward. Cees has become a good friend, who encouraged and supported me in doing this work based on his broad vision and enormous dedication to research.

Prof. dr. Robert Meijer, with whom I had many inspiring discussions on new and challenging ways to consider and apply different concepts and technologies involving programmable network paradigms. His inspiration evolved into several new and innovative ideas, essential in taking my work forward.

John Vollbrecht at ADP Network Services, MERIT, Interlink Systems and Internet2. Since the start of my career in 1981, John has always been there for me as a great help giving me guidance and wisdom. Over the many years I have had the honour to know John as a friend, he always inspired me to perform research on topics that we both wanted to pursue. As such, we took great challenges on our shoulders. At first it was the topic of Authorization and later the topic of Trust after he retired. We had many lengthy discussions during which I am very grateful for John's way to ask questions that gained many new insights and also for his patience during the process.

Dave Spence at ADP Network Services, MERIT and Interlink Systems. Dave was a great help during our discussions around AAA and later on Trust. Dave's ability to scrutinize concepts on its consistency was amazing and always a great help.

Sue Hares at ADP Network Services and MERIT. Sue helped me when she chaired the AAA BoF meeting in Chicago in 1998. This meeting was the starting point of my research. Her work on QBONE, the Bandwidth Broker work, she happily shared with us, was a great source to get our work started.

At SURFnet: Kees Neggers and Erik Jan Bos, who supported and believed in the importance of this research topic by granting funding via the GigaPort projects. I thank Kees and Erik Jan in particular for sharing thoughts and insights into the NREN world. Also I like to thank Klaas Wieringa for helping me better understand the world of roaming.

At Internet2: Steve Wolff for supporting our research efforts performed on optical networking, AAA and trust from his end.

At BBN: Chip Elliot for his feedback and encouragement to pursue the subject of Trust in the modern world-wide cyber infrastructure world.

At Cabletron Systems: Foremost CEO Craig Benson, who initially allowed me to work with the research world at University of Utrecht. CTO's Mike Skubitz and John Roese, by recognizing its importance. The twin Dobbins brothers Kurt & Kris and Paul Lachappelle, with whom I worked on AAA concepts in the context of virtual networking and discussed idea's to implement them.

In various project efforts, where I felt the honour of being supported by: Henri Bal, Bob Hertzberger, Wim Lourens, Tom DeFanti, Maxine Brown, Joe Mambretti, Bill St. Arnaud, Fabrizio Gagliardi, Larry Smarr, Gigi Karmous Edwards, Franco Travostino, Inder Monga, Rodney Wilson, Admela Jukan, Dimitra Simeonidou, Arthur Binczewski and last but not least Bernhard Fabianek, our EU project officer, who all helped by pushing my AAA research topic forward. They also helped me to put the context of my research in a broader perspective.

All co-authors, UvA SNE group members and co-researchers with whom I had the pleasure to work on publications with and have not been mentioned so far: Yuri Demchenko, Bas van Oudenaarde, Arie Taal, Fred Wan, Freek Dijkstra, Martin Hoekstra, Mihai Cristea, Rudolf Strijkers, Jeroen van der Ham, Paola Grosso, Tal Lavian, Zeger Hendrikse, David Groep, Oscar Koeroo, Rene van Buuren, Olle Mulmo, Martijn Steenbakkers and Andrew Tokmakoff

All the people I worked with and without whom our experiments would not have been successful: Jeremy Weinberger Madhav Srimadh, Satish Raghunath, Paul Daspit, Bruce Schofield, Chetan Jog, Phil Wang, Fei Yeh, Pieter de Boer, Hans Blom, Dennis Paus, Bram Peters, Phil Wang, Satish Raghunath, Fei Yeh, Thomas Tam, Tuan Nguyen, Tom Lehman, Andrew Lake, Jarda Flidr, Brian Cashman, Eric Bernier, Bert Andree, Ralph Koning, J.P. Velders, Jeroen Roodhart, Sumit Naiksatam.

At CERN: Olivier Martin, head of the networking group, with whom I had good discussions on the rationale of by-passing the regular internet during the DataTAG project.

At NIKHEF: Arjen van Rijn, who helped us get involved in collaborative High Performance and Grid Computing efforts that was initially lead by the High Energy Physics community via projects such as DataTAG, DataGrid and EGEE.

At INFN and OGF: Tiziana Ferrari, with whom I worked on idea's around QoS networking.

At AMS-IX: Henk Steenman and Job Witteman for providing essential insights into the essence of the role of an Internet Exchange.

At SARA: Jules Wolfrat, Andree Toonk and Ronald van de Pol, who patiently helped us with arranging our experiments and demonstrators.

At Intel, Christophe Diot for sponsoring our research by providing a network processor development platform.

At VU Amsterdam, I am grateful to Herbert Bos for providing guidance with microprogramming network processor platforms for our experiments.

At International Card Services, Alex Bewier for his constructive feedback on the abstractions describing Payment Card Services.

At Sinds1961, Sharon Wijnhoud for her help and patience during the layout process of this thesis.

At IETF I am grateful having worked with George Gross, Steven Farrell, Matt Holdredge, Pat Calhoun, Nevil Brownlee, Larry Dunn, Erik Huizer, Bert Wijnen, Bernard Aboba, Fred Baker, Brian Carpenter, Donald Eastlake 3rd, Dave Harrington, Shai Herzog, Paul Krumviede, Brian Lloyd, Charly Perkins, John Strassner, Walter Weiss, and many more.

At OGF I am grateful having worked with Ian Foster, Charly Cattlet, Carl Kesselman, Steve Tuecke, Markus Lorch, Thijs Metch, Dave Chadwick, Bill Johnston, Mary Thomson, Dane Skow, Krishna Sanka, Lavanya Ramakrishnan, Andrew McNab, Paul Madsen, Rich Baker, Bob Cowles, Ralph Niederberger, Bill Allcock, Egon Grunter, Gian Luca Volpato, Christian Grimm, Guy Roberts, Tomochiro Kudoh, Jerry Sobieski, Alan de Smet, Jenny Schopf, Steve Newhouse, Frank Siebenlist, Pascale Primet, Richard Hughes-Jones, and many more.

At Air France-KLM, I am grateful to my subsequent managers, Timor Slamet and Kees Wijnstra. I am also grateful to Taco Brouw and Hans Kooiman, the subsequent department Vice Presidents. They all allowed me the freedom and opportunity to work on this thesis and supported me all the way. I am also grateful to KLM for being such a great employer.

But above all, I like to thank my dear wife Betty. She not only contributed to this work regarding the topics of Authorization and Trust from her elaborate payment world perspective, but she also supported me with her caring love and above all her patience. I owe her a lot of respect as, without her understanding, this work would not have been possible.