An Entity Access Control Model for Network Services Management

June 2005

Bruno Dias
Departamento de Informática, Universidade do Minho
Campus Gualtar, 4715-057 Braga, Portugal
<bruno.dias@di.uminho.pt>

Abstract

The Network Services Management Framework[i] tries to overcome the most important limitations of present network management frameworks, namely the most widely supported framework – the Internet Network Management Framework – by defining a management framework using a network services management distributed architecture that provides services management functions with any desired level of functionality. This document introduces one of the most important parts of this framework, the Entity Access Control Model and the mechanisms needed to its deployment: management entities and management domains, entity access and resources control management, and security mechanisms (authentication, data integrity verification, confidentiality and non-repudiation assurances). This model, although originally developed to be integrated on the Network Services Management Framework, can be completely integrated or partially adopted by other frameworks since it supports a wide range of conceptual and functional requisites recognised to be fundamental to the future of modern distributed network management frameworks.

Keywords: Network Services Management, Entity Access Control, Management Entities/Domains, Resources Control, Security Mechanisms.

1 Introduction

Although there's been a continuous evolution in Internet Network Management Framework (INMF),[ii] the most recent achievements been the group of documents that form its third version (INMFv3) [4,5,6,7,8,9,10,11,12] and the second version of the Remote Monitoring Management Information Base (RMONv2) [13], it has been is recognized [1,2,20,21,23] that this framework is not providing what his users (network administrators using SNMP based devices and management applications) demand: an efficient mechanism for global management of modern Network Services. Some of these problems have been addressed in the past years with the creation of mechanisms that try to increase the functionality level of the managed objects and make the model less centralized. Such efforts include the distributed management [22,23] and management by delegation [22] concepts that are partially and indirectly implemented in a variety of INMF extension MIBs [14,15,16,17,18,19]. Nevertheless, the functionality level of the individual objects of these MIBs is still very limited and their management, from the managers point of view, complicated and resources consuming.

In the approach defined on the Network Services Management Framework (NSMF) [1,3], formerly known as Internet Network Services Management Framework (INSMF) [2], these concepts of distributed management and management by delegation are natively implemented using the same integrated mechanism and can be provided directly to the user as Services Management Functions (SMF) and is intended for the management of network services and distributed applications. If needed, SMFs support the concept of MIB objects as information management resources [2]. Also, a more complete and flexible management architecture matured from the initial definition proposed on the INSMF [2] with the evolution of the generic management entities concept and their integration on administrative and

[i] This management framework was originally named Internet Network Services Management Framework due to the fact that it was a first result of the scientific investigation on how to improve the standard Internet Network Management Framework. By now, this project has evolved to a more general and complete network services management framework and hence the new name.
[ii] Also known as the Simple Network Management Protocol (SNMP) framework.

functional hierarchic management domains. So, the most relevant and recent concepts created for or introduced in network management are natively incorporated into the NSMF model.

2 The NSMF Model

The NSMF architecture [1,3] has become an independent and generalist management architecture, that is, it is a much broader management architecture when compared with the INMF and other major network management frameworks, permitting any level of management functionality, from a higher level of abstraction (with a rich set of semantic definitions) to a lower level of abstraction (with much narrowed semantic definitions, like the one found on the INMF); it doesn't require the INMF nor replaces it. In fact, it's a pure functional framework not bound to any management information model and tries to integrate the most important and recent concepts on network services management studied, developed, or introduced (with more or less adaptation) on any management framework with some repercussion on the computer networks community. Although being an engineer's project, the development of the NSMF does not have the commercial or fast implementation concerns and constraints of the INMF and other widely deployed management frameworks. By now, the NSMF project main inspiration is to provide the most complete (in terms of range of network services management functionalities) and flexible (to permit future evolutions and integration of new mechanisms with easy and without the need to define new versions of the framework) network services distributed and highly hierarchic management framework. It has no politic constraints nor it is tied up to a complete network protocol framework and has started as an effort to take over some of the more important limitations of real network management frameworks (in the sense of widely implemented/available, like the INMF) as they are perceived by real network administrators and network management application software developers. Finally, the NSMF project major goal is to provide to the computer networks community a complete, independent[i] and flexible network services management framework. It is like a reference document on how to build and implement network services management frameworks. Specific management frameworks could use only the concepts and mechanisms needed to match their pre-defined functional and conceptual requisites.

Conceptually, the NSMF complete definition is divided in thee parts:

1. Entity Access Control Model – The NSMF architecture is supported by a generic Network Services Management Entity Access Control Model, or just Entity Access Control Model (EACM), when it comes to the definition of rules on how to build Management Domains, how to identify Management Entities and what are the requisites for the mechanisms to ensure secure communication between entities. While the EACM creation was triggered by the need of an entity access control mechanism for the NSMF, the resulting broader definition can be employed on other frameworks. In fact, for the EACM to be applied on the Internet, some particular definitions are made and named as default choices, meaning that the default usage for this model is on a network management framework for the Internet. Other particular choices can be made on some of the EACM components, making it more useful for other types of network architectures, even architectures other than management services models. This document main intention is the presentation, with some detail, of this NSMF architecture major component.
2. Protocol Interactions and Data Units – This part of the framework defines the methods that the management entities have to communicate to each other (that is, to send and receive SMF execution requests and SMF responses) using the mechanisms defined on the EACM. The defined protocol interactions provide independent and generic management information transport mappings, with capabilities for both connectionless and connection-oriented communication.
3. Services Management Function – The SMF was, initially, the most important single concept of the NSMF, which is a procedure implementing a group of well defined services management actions/functions, including the execution of other SMFs. A set of SMFs can be grouped or created together as a library of SMFs – a SMF Definitions Base (SMFDB) – if and when they relate to the same management service and have similar functionality levels. One SMF code can request the conditional and/or scheduled execution of other individual

[i] The NSMF does not need the integration of other network services to provide management services. For example, the NSMF applied on the Internet does not need to use the Domain Name Service and the encapsulation of SMFs can be made using a connectionless management communication or connection oriented management communication independently of the data transport protocol used or the type of network and application addresses. Also, security is provided without the need of any external security service.

SMFs or complete management procedures, since the NSMF has native support for code delegation. Furthermore, there is also native support for a wide range of network services monitoring techniques due to the possibility for results inspection during the execution of the SMFs.

3 Entity Access Control Model

The most important requisite for an entity access model that could be used on the NSMF is the capability of implementing mechanisms for conversion between entities names and addresses and for registration and delegation/distribution of entities profiles (set of parameters needed to inform the other entities about ways to establish secure management communications) without the need for external network services, other than the chosen network/transport protocols. Since we were defining an access and control model for use by management architectures, it would be essential to maintain such architectures independent of the use of other application network services that would risk the functionality of the management services it selves.

The EACM is based on some existent access control and security models, like the User-based Security Model [8] of the SNMPv3, Internet network application services like the Domain Name System (DNS) and user access and resources control mechanisms employed on some of the most common operating systems. Another important pursued virtue is the simplicity of usage. While the model should be able to incorporate a broad range of mechanisms (access control, resources management, key registration and distribution, etc) and accommodate the NSMF need of implementation of a wide spectrum of functionality levels through the use of Service Management Functions (SMFs), the model should be defined around simple and generic mechanisms, independent from each other and flexible enough to permit different types of particular implementations, depending on the particular architecture where it would be incorporated.

The concepts of Network Services Management Entity, Network Services Management Domain and Network Services Management Domain Registration and Information Server are the key elements on the EACM architecture. There are four types of entities defined on the NSMF model that can implement and/or use SMFs:

- Management Service Agent (MGT-SA) – entity implementing a minimum group of SMFs but only uses internal or external management information resources available by means different than SMFs;
- Management Service Manager (MGT-SM) – entity issuing SMFs calls on one or several MGT-SAs (and/or MGT-SSs) as a mean to implement network services management procedures needed to implement the network productive protocols;[i]
- Management Service Server (MGT-SS) – entity acting like a MGT-SM and that, additionally, can provide an interface to its implemented network services management procedures as SMFs. That is, a MGT-SS issues SMFs calls on MGT-SAs (or other MGT-SSs) when implementing his SMFs that should have higher levels of functionality; and
- Management Service Application (MGT-SAP) – entity issuing SMFs calls on MGT-SAs or MGT-SSs as a mean to implement network services management procedures not defined as SMFs, but as part of a foreign management framework or part of an implementation of a management application.

These entities can reside on any networked equipment. For example, a typical internet router can host several MGT-SSs and MGT-SAs, each implementing one or several network services management. The necessity for the simultaneous existence of various MGT-SSs or MGT-SAs on the same equipment has to do with the fact that different access and security policies can be used, depending on the service being managed. More generally, the term management server (or just server) will be used to identify an entity that implements and provides some kind of network management service and management client (or just client) will identify an entity that issues SMFs on the management servers.

This model defines:

- Entity and Management Domain Identification rules – while this model imposes a syntax for entity and domain naming (defining a double naming hierarchy) it only defines a general

[i] The network productive protocols or mechanisms are all the protocols implemented on the network apart from the protocols or mechanisms needed solely by the management framework.

entity address syntax, although a default/particular address ruling definition to use on the INSMF is given;

- Management Domain Registration and Information Service (MDRIS) – this management sub-service provides a way for management domains and entity's profiles to be securely registered, distributed and accessed when needed; these profiles define capabilities of the management entities, the parameters of the adopted security mechanisms and the level of access permitted for each domain and entity; this management service is provided by a special type of MGT-SA, named Management Domain Registration and Information Server;
- Management Communications with Resources Control – this model defines secure management communications between management entities, with the capability of dynamically limit the functionality of management procedures or the consumption of network and local device resources depending on the entities involved and the state of the network and the devices during the time of execution of the management procedure;
- Functional requisites for the security mechanisms to be used on the model with additional definition of default mechanisms for implementing authentication, confidentiality and data integrity on management sessions. One of the aspects of this model is the use of default security mechanisms based only on private symmetric keys. The distribution of entity profiles, including their security parameters and secret keys, is based on a trusted registration and distribution process. Each entity only needs to register its profile[i] on a trusted, already registered, entity belonging to the same domain. This trusted entity is a Management Domain Registration and Information Server (MDRS) and will distribute the entity's profile to the other MDRSs of the domain, so any of these MDRSs should be able to provide the needed information when another authorized entity requests that profile information for, for example, opening a management session.

3.1 Entity Identification & Addressing

The identification of the NSMF entities is based on a double hierarchic domain structure, each one resembling the hierarchic structure used on the internet's domain and host naming. But, unlike the internet generic names, these NSMF names are of mandatory use, that is, the identification of a management entity is only possible through the use of its NSMF name, so, each entity will have, at least, one NSMF name – a sequence of string labels. Furthermore, while a particular naming syntax is defined, following a rigid set of rules for naming management entities and domains, that is of mandatory use on the EACM, there is only the definition of a configurable syntax for entity's addresses and, at present, as an illustration example, a particular syntax definition is proposed for use in the NSMF when using the User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) over the Version 4 of the Internet Protocol (IPv4).

Management entities are grouped in Management Domains. The management domains names and entities names don't need to relate to internet domain and host names, but management domain names can derive from internet domain names (or any other type of network entity names, depending on the type of network where the management service is to be deployed). Each NSMF entity name is a set of string labels divided in two parts: the entity prefix name (identifying the entity inside the management domain) and the management domain name.[ii] For example, an entity name of atm.nsmgt.uminho.pt has a prefix name equal to atm and a management domain equal to nsmgt.uminho.pt. The entities names and addresses, and all the information[iii] needed on the EACM, are registered on special entities – the Management Domain Registration Servers – dedicated to the task of storing and providing this information to other

[i] Besides this Entity Access Control Private Profile (EAC-PP), each entity must register its management name and address. Included in the profile is the set of its private keys, or information on how to derive the values of these keys.

[ii] Each string is a sequence of one or more characters, with each character being represented on a byte value (254 values can be used – two values are for reserved application). The visual representation of these EACM entities names dependent on the values of the characters and on visualization rules to be applied to them. The visualization of entities or domains names is controlled by an additional field included on the syntax of the EACM name, the Display Hint. So, what it is displayed depends on the display hint of each name and on the capacities of each management application to display the name following that hint.

[iii] All the information present on the Entity Access Control Profile is to be registered on the Management Domain Name Servers.

management entities. These servers do provide all the mapping between entities names and entities application addresses (including the IP address if the NSMF is implemented over Internet Protocols) without the need for any external network service or protocol, making it more reliable on troubled networks where a mapping service could become unavailable. Also, the EACM model definition should be generic and independent of any existing particular framework, protocol or application.

A management entity's address must completely identify a management entity implemented on a network device.[i] Additionally, the address value will indicate the transport service/protocol used to convey the management communications between two entities. As a general definition, a Network Services Management Access Point (NSMAP) is an application process running on a management domain implementing a network services management entity. A NSMAP address must identify the application process executing on the network devices of the management domain and the network transport service/protocol to be used to encapsulate the management protocol data units. Since an entity can have several addresses, it is possible for an entity to use more than one network transport service/protocol, either on the same framework, like Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) over IPv4, or TCP and UDP over IPv6) or on network transport protocols of different frameworks. This addressing flexibility and the native support for connectionless management communication messages and connection oriented management communication sessions to encapsulate the NSMF protocol units, it is possible to use network protocols on lower levels of network frameworks, like IPv4 or IPv6, or even link level protocols (although no communication between entities on different data links on networks that do not provide routing on this level would be possible). It is possible that an entity address points, indirectly, by means of an address translation table, to a third entity on another network device. The original destination entity (server process) is defined as a Proxy Entity and the final destination entity (also a server process) is defined as the Target Entity. The address mechanism defined on the EACM provides means to implement proxy addressing with many levels of indirection, that is, a proxy entity's address can point to another proxy entity's address, and so on, until a final target entity is referenced. This local mapping[ii] (or proxying) capability of the EACM can be useful to distribute network services management between several network devices, based on local distribution policies that should take into account the nature of the management service, the resources available on the local distributed management services network, the local security policies and other administrative policies. Further, this local management services network distribution policy can be dynamic (even using other management services to make dynamic distribution decisions) or following some fixed/static distribution rules. Thus, this address translation mechanism is a way of implementing distribution of management tasks and/or dynamic addressing without the need to frequently change the registered name and address of entities implementing a particular network management service.

3.2 Management Domains

A Network Services Management Domain represents a group of management entities implementing a set of SMFDBs dedicated to a common network service management. More generally, it will identify a group of trusted entities that should have something in common, some kind of functional or administrative affinity, and will be registered on the same Management Domain Registration Servers (MDRSs).

The names used to identify management domains are divided in two sections: the Network Services Management Domain Identification (NSM-DI) and the Network General Domain Identification (NG-DI). Each sub-domain prefixed must identify a sub-group of entities with a particular type of management functional affinity between them. This affinity should be related to the type of management services provided or since it is recommended that each NSM-DI domain tree – starting by the nsmgt string on the example of Figure 1 – relates to a group of management domains all with the same network administration or sharing a common network administrations hierarchy. The NG-DI part is like the standard Internet DNS part but can be null. This part should be divided on a network administration authority base, probably following the same DNS structure. The hierarchic order is defined, separately, on two levels, being the NG-DI part the most significant, that is, the NG-DI part defines the primary order of the domains and the NSM-DI part defines the secondary order inside the same NG-DI. The hierarchy on the NSM-DI part is naturally implicit, that is, the creation of a NSM-DI sub-domain is only valid and useful when that sub-domain is registered (and thus *linked*) to its upper level domain. On the other hand,

[i] A management entity can be implemented on one or several network devices, but it will be seen as a unique point of access for a management service client or server.

[ii] Local here refers to the fact the translation mechanism must be implemented locally on each proxy entity.

there is no obligation for management domains with different NG-DI parts to be linked to each other, even if the NG-DI parts of the name represent a hierarchy. A Network Services Management Domain Hierarchy (NSM-DH) is the group of domains naturally linked. It is possible to link two management domains that are not part of the same NSM-DH by registering each other's profile on the MDRSs of the other partner domain. When two management domains form a partnership, the two NSM-DHs also become linked, although the link is only useful to the entities on the directly linked domains or on lower level domains of both NSM-DHs.

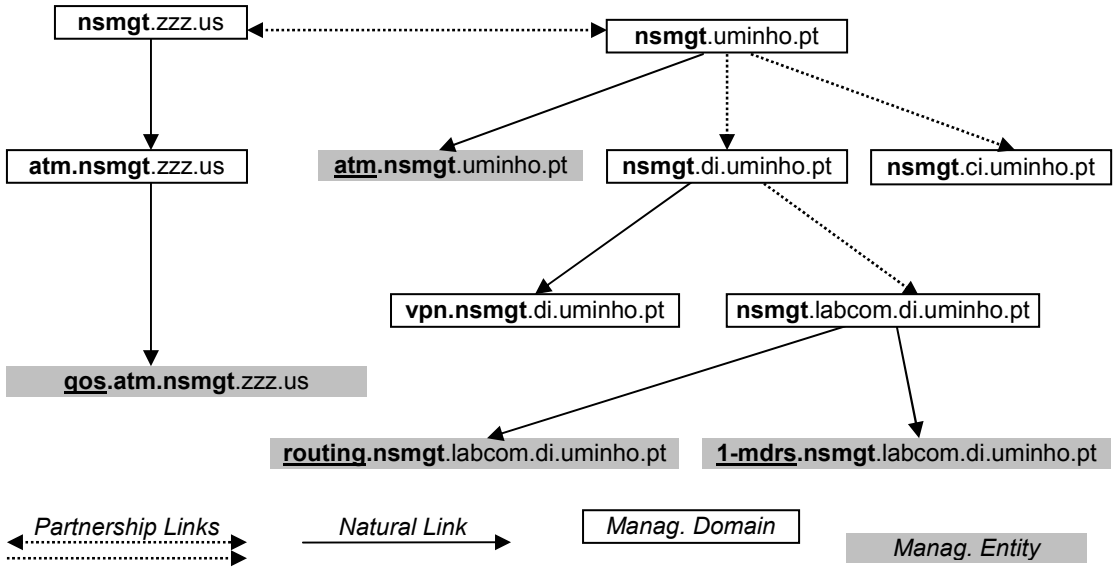

Figure 1: Example of EACM naming.

3.3 Management Domain Registration & Information Service

This service is part of the EACM model and is provided by the MDRSs of a management domain and must support the SMFs defined for this special network services management service that implement management procedures for:

- Domain & Entity registration – this group of management functions are used to maintain a synchronized image of the profile of the management domain and all the entities across all the MDRSs of that domain; this will include a group of management procedures implementing functions for registration, maintenance and deletion of entities profiles (the profiles will include management names, management addresses, resources access control levels, etc);
- Definitions of hierarchic domains relationships – this group of management procedures include functions to register relationships between management domains;
- Security information registration and maintenance, that is, a key management service – this group of functions provides means to register and maintain lists of dynamic security keys/secretes and any security data needed to deploy the supported security models on the management domain; it must be noted that the MDRSs are, from a security point of view, trusted third parties;[i]
- Accessing information about management domains and management entities – this group of functions are used to query the MDRSs about domain's or entity's profile information including security information needed to the establishment of secure management communications between generic management entities; some of these functions can be used across management domains.

3.4 Resources Control

One of the most important aspects of the EACM is the capacity for defining levels of access to the various types of resources available on a network device or on the network. These levels are defined by default on a management domain basis, but can be redefined by each entity acting as a management server, both by adding support to other local resources types or supporting additional access control levels besides the

[i] Being a trusted third party, the MDRSs provide the means, through the MDRIS, to support an optional non-repudiation security mechanism to both management clients and management servers.

domain default access control levels.[i] The definition of access control levels takes into account the type of resources being used by each SMF issued by the management clients on the management servers. Unlike the access control of MIB objects on the INMF, for example, that associates a single fixed level of access (from a very limited set of available levels) for each MIB object to a SNMP manager, the access control on the EACM associates a single dynamic level of access (from a potentially infinite range of levels) for each type of resource (from a potentially infinite range of types of resources) to a EACM management client. Table 1 lists the most important standard EACM types of resources arranged on a two-level hierarchy organization.

Type of resource	Sub-type	Tag
Memory Resources	Code on Volatile (%)	Memory.code
	Data on Volatile (%)	Memory.data
	All Volatile (%)	Memory.volatile
	All on non-Volatile (%)	Memory.external
	All (%)	Memory.all
Network Resources	Bandwidth (%)	Network.bandwitdth
	Buffers (%)	Network.buffers
	All (%)	Network.all
Process	CPU Load (%)	Process.CPU
	Number of processes (%)	Process.quantity
	General (%)	Process.all
	Priority (%)	Process.priority
Profile Configuration	MDP (Enum.)	Profile.MDP
	MDRS (Enum)	Profile.MDRS
	Private (Enum.)	Profile.private
Logs	NS Management (Enum.)	Logs.management
	IP protocols (Enum.)	Logs.ip
	Routing (Enum.)	Logs.routing
	Users (Enum.)	Logs.users
	Hardware (Enum.)	Logs.hardware
	Other Logs (Enum.)	Logs.other
General Configuration	NS Management (Enum.)	Configuration.management
	Link Level Protocols (Enum.)	Configuration.link
	IP Protocols (Enum.)	Configuration.ip
	Network Services (Enum.)	Configuration.ns
	Applications (Enum.)	Configuration.applications
	Other Protocols (Enum.)	Configuration.others
	Hardware (Enum.)	Configuration.hardware
	General (Enum.)	Configuration.general
Delegation	Code (Enum.)	Delegation.code
	Results (Enum.)	Delegation.results
	General (Enum.)	Delegation.general

Table 1: The most important standard types of resources.

On the other hand, the access levels values can be of two types:[ii]

> Percentage – All quantitative types must be abstracted as a percentage value (and some extra special values also). These values represent a quota that must be guaranteed to the management clients when they issue SMFs on the management server, although the actual consumption of the resource at a specific moment could be less than the quota defined on the granted access level. This value can have any desired resolution and some of its values are reserved for special meanings. Some of them have already been defined as standard on the EACM and are listed here:

[i] The management clients can negotiate further access levels for each particular management communication with each particular management server.

[ii] It should be noted that an access resource type and sub-type identifies the concept/semantics of the resource (memory, configuration, etc), while the access resources levels type identifies the syntax of the values used to represent the specific levels of the resource type/sub-type.

  - min (101%) – default minimum percentage value for usage of the resource on each particular management server; the actual value depends on that management server;
  - max (102%) – default maximum percentage value for usage of the resource on each particular management server; the actual value depends on that management server;
  - any (103%) – usage with no restrictions, that is, the management communications issued by management clients that can obtain this access level, will be able to use all the quantity available of the resource on the network device where the management server implementation resides; this level of access means that the processes created on the management server to execute the management procedures issued by the authorized management client will take, if needed, all the available resource on the device, even beyond the reservation level; it must be noted that this value of access level is particularly dangerous and should only be allowed on some types of SMFs issued by trusted management clients;
  - avg (104%) – default access level for each particular agent, that is, the management clients obtaining this access level when registering, will have the local default access level on each agent on the domain, if applicable; the actual value is defined on each management server;
  - shr (105%) – shared access level, that is, the management clients obtaining this level of access will dynamically and equally share the remaining quantity available on the resource on the server; if the management server is executing only processes from clients that obtained this access level then each client (not each process) will have an equal share of the reserved level of the resource; if the server is executing additional processes issued from clients that obtained a fixed access level (or quota), the resource occupied by the processes from these clients will be deducted from the quantity available to the former management clients.
- Enumeration – This type of access level value is defined as a set of enumerations. Each enumeration will be represented as a flag (coded as a bit value) and the various levels are not exclusive. An entity's profile could define keys for obtaining several access levels at one time, besides the individual definition of keys for each level. The values defined depend on the type (and sub-type) of each particular resource. Since the values are not exclusive, one bit for each possible value must be used.

On the EAC-BC…

Resource	Access Level	Key
Memory.all	100	ksh46dks
	min	(1)
	max	4s2jfg34
	any	987fhf92
	avg	hrtfs5sq
	shr	fjz54cc8
Network.all	min	(1)
	max	8umsl35t
	any	jfd5pafd
	avg	9318394g
	shr	zvx42xfd
Profile.MDP	synch	h57rfhdd
Config.ip	status	(1)
	monitor	07r5322z
	conf-int	pwsk5fde
	reset	yff87hr4
...		

(1)*Default, no key required.*

On the management server's profile…

Resource	Access Level	Key
Memory.all	min	(2)
	shr	(2)
	any	j8y87uyy
Config.ip	status	(2)
	conf-all	0rr63y29
	reset	(2)
...		

(2)*Assumed from EAC-BC, no key required.*

On the management client's profile…

Resource	Access Level	Key
Memory.all	any	j8y87uyy(3)
	shr	fjz54cc8(3)
Config.ip	monitor	07r5322z(4)
	conf-all	7p4yr64t
	reset	yff87hr4(3)
...		

(3)Access level obtained for the depicted server.
(4)Access level obtained for the domain but not for the depicted management server.

Table 2: Examples of resource access control levels definitions.

Table 2 gives a partial view of the access control levels definition lists contained on a default management domain profile, or Entity Access Control Basic Configuration (EAC-BC), on one of its management servers and on one of its management clients. The EAC-BC defines for resource type Memory.all six access control levels, with a default value of min, that is, all the management clients on

the domain will obtain automatically this access level on the domain.[i] It can be seen that other resource types have also a default level defined. Also, except for the Profile.MDP resource type that only has one level defined, there are additional levels and their associated keys. The depicted excerpt of the management server profile defines three access levels for Memory.all and another three levels for Config.ip. Each of these resource types has two levels with no key defined, which means they inherit the key values associated with the same levels on the EAC-BC, if defined. If these levels are not also defined on the EAC-BC any management client on the domain will obtain automatically these access levels. For access levels that have associated keys on both profiles (EAC-BC and management server profile) the definition on the management server profile takes precedence, that is, even if the management client obtains the access level for the domain by matching the key value on the EAC-BC, the access level for this specific management server will only be obtained if the local key value is matched by the management client.

3.4 Security Model

The security features of the EACM were defined taken into consideration the most traditional security threats relevant for management frameworks: masquerading, information modification, deletion or disclosure. Although other security threats could be identified, these have not been considered relevant to network services management (like traffic analysis or denial of service) or can be reduced to a particular or combined form of the previous threats (like information sequence or timing modification). On the other hand, even if authority repudiation is considered to be a lesser threat to network management, it would be useful that a management framework could provide an optional security mechanism to be used where would be important to guarantee non-repudiation. Being so, the EACM supports an optional mechanism that can be imposed either by a management entity acting in a role of a management server or an entity acting in a role of a management client.

The most important goals of the EACM concerning security can be briefly listed: guarantee of entity authentication and management information confidentiality, verification of correct sequencing and integrity of management information. Another important goal is the ability to support access control to network and entity resources. Although less important, it is an objective of the EACM to provide the means to natively support non-repudiation assurance. These goals should be attainable using a set of pre-defined mechanisms, identified by an EACM Security Model Identification Tag (SECM-IDT) and applicable to each Management Communication between two management entities. While an entity can support several security models, only one security model can be applied to each management communication. It becomes obvious that both entities involved must support the security model applied to that management communication. At least, management entities must implement the default management domain security model to be able to communicate to the MDRSs on their domain.

The security mechanisms to be applicable on the EACM must not use external protocols or security mechanisms.[ii] These mechanisms must be defined and implemented inside the NSMF so it stays an independent framework with no external protocol, mechanism or system dependency. Further more, the security model of the EACM must define a set of functional target goals for each mechanism and the protocol syntax constraints imposed. This way, several present and future alternatives can be used for each one of the mechanisms for:

- Encryption – the EACM needs an encryption method to be applied to the appropriate[iii] part of the NSMF PDU so confidentiality can be supported. The standard encryption mechanisms defined, at present, to be applicable on the EACM are based on traditional symmetric key encryption methods, being the most important the Advanced Encryption Standard (AES) [27] – this is the most scrutinized cryptographic standard of the last years, based on the Rijndael symmetric block cipher algorithm.[iv] An important issue is that it is believed that this mechanism does not shows weak or semi weak keys or any type of key restrictions, it is

[i] This access level will only be meaningful on management clients that implement it.
[ii] External here means protocols, mechanisms or systems, defined and implemented outside the NSMF.
[iii] The appropriate part will be the section of the NSMF PDU data stream subject to encryption defined as the NSMF PDU Encryption Target (PDU-ENC).
[iv] In true, some important work has being done in the last two years proving, at least in theory, that the AES mechanism is weak to algebraic attacks [28,29] (although it was not broke until now) and that it should only be used when no long term security is needed, which is the case of its application on the EACM. Nevertheless, it is considered a major step ahead in comparison to the majority of the widely available symmetric key encryption mechanisms, namely the Data Encryption Standard (DES).

immune to linear cryptanalysis and differential cryptanalysis and it is a relative simple algorithm permitting implementations with fast execution times [26]. Since the AES permits three key sizes (128, 192 and 256 bits), the EACM will be able to use these three versions, identified as three different encryption mechanisms: AES-128, AES-192 and AES-256. There is also default support for Triple Data Encryption Standard (3DES) [30,31] – that is still recognized as a relatively secure and simple mechanism to implement, although slow, encryption mechanism [26]. Other methods can be applied (Serpent, Twofish, RC6, Mars and Saffer++) but are not defined as EACM standards at this time.

- Keyed Message Digest – this is needed to ensure authentication (by itself or in conjunction with the encryption method) and verification of data integrity of the appropriate NSMF PDU part on the EACM. This type of mechanisms became, in the last decade, the most used security methods for message/data authentication and integrity verification because the message authentication is obtained without exposure of the key/secret and they are open standards that can be widely used without any type of copyright limitations. They can be based on non-keyed message digest algorithms, like the Key-Hashing for Message Authentication (HMAC) [33,34,40] mechanism or can be based on message digest compression techniques from symmetric key encryption mechanisms, like DES-Message Authentication Codes [32]. It was decided to define only the former type of mechanisms as standards on the EACM. The most important advantage, in respect to the later type of mechanisms, is that they are based on well known non-keyed message digest algorithms that are still believed to be more secure[i] and efficient, with lower resources consumption [26]. The EACM adopted the HMAC mechanism with the possibility to use three well known message digest methods: Message Digest 5 (MD-5) [33] (although some weaknesses were found on MD-5 [36] is still believed that this mechanism can be securely used on HMAC); RIPEMD-128[ii] [30,31] (other versions of RIPEMD, namely RIPEMD-160/256/320 [38] could also be used as message digest algorithms inside HMAC); Secure Hash Algorithm 1 (SHA-1) [39] (SHA-1 is believed to be securely stronger than MD-5, although a slower algorithm, so it is preferred for application on the EACM). The keys for use with HMAC must be strong (closest to random as possible) and must be, at least, 80 bits long. Good randomness of the keys should be guaranteed by the EACM key renovation mechanism and the recommended minimum key size for standard security models of the EACM is 128 bits (16 bytes).
- Key Renovation & Distribution – the EACM must have a well defined mechanism for renovation (creation and deletion) of the keys/secrets of the management entities registered on management domain. In direct relation to key renovation is the need for a method for key transfer (distribution) between management entities. The EACM provides standard support for two key renovation approaches:
    - Local Renovation – The keys are only created (or renovated) locally on the entity owning the keys and then sent encrypted to one of the MDRSs of the domain.
    - Distributed Renovation – The keys are created (or renovated) at the same time by the entity owning the keys and by the MDRSs through the use of a pre-defined mathematical process. This mathematical process should be supported on the domain (at least by the MDRSs and the respective entity) and indicated as an argument to a special SMF, defined on the Management Domain Registration & Information Service, executed on the MDRS and issued by the entity owning the key. In simpler terms, this specific SMF uses some present/old key values and a stream of bytes chosen by the key's owner to generate one (or several) new keys. The requisites of the stream of bytes are dependent on the referenced algorithm for key renovation. The standard algorithms require that the stream of bytes be as close as possible to a random sequence of bytes and of a sufficient length. The owner entity must also execute the same renovation SMF locally (with the same arguments) so the resulting new key value(s) is (are) equal to the resulting value(s) on the MDRS. This approach is safer because the keys values are not exposed (even if the previous methodology uses encryption) since they are not on transit on the network during the renovation procedure. At this moment, the EACM provides definitions for use of three standard algorithms for key renovation based on

[i] Although some weaknesses have been found on MD5 [35], they don't appear to have an important impact on its usability on keyed message authentication [34].

[ii] The acronym RIPEMD stands for RIPE Message Digest and RIPE means RACE Integrity Primitives Evaluation [37]. Finally, RACE stands for Research and Development in Advanced Communications Technologies in Europe.

three flavors of the HMAC mechanism: HMAC-MD-5, HMAC-RIPEMD-128 and HMAC-SHA-1.

- Data Compression – there can be an optional use of a lossless data compression mechanism that will help to minimize network bandwidth consumption and, if applicable, to reduce the security vulnerability of the original data stream of the appropriate NSMF PDU part/section. At this moment, the EACM defines only one data lossless compression mechanism, based on the Huffman compression algorithm concept: digits of an alphabet that appear with higher frequency will have codes with fewer bits (smaller codes). Since several algorithms exist based on this traditional approach, the EACM only defines the syntax rules of how to represent the code table to be applied by the entity performing the decompression. This way, an added flexibility is achieved and the entity implementing the decompression does not need to know exactly which algorithm was used to construct the table of codification. It will be up to the entity making the compression to adopt the compression algorithm that better suits its goals (speed of compression, rate of compression, resource consumption, etc) as long as it is a lossless method and the compression codification table dos not yield any ambiguity and follows the syntax rules defined by the EACM.
- Non-Repudiation Assurance – finally, the EACM should provide, although optionally, mechanisms for assurance of non-repudiation. The NSMF supports an optional security mechanism that can guarantee non-repudiation using any MDRS on the management domain as the third party, trusted by the other entities on the management domain. The most common situation where the use of such mechanism is useful is when a management server wants to ensure that a management client can be identified as the responsible entity for an executed management procedure that resulted on some kind of functional constrain, even if the client had permission to do so. The management server is able to report the authority of the management procedure since the mechanism defined on the EACM guarantees that the reported client is the only entity (non MDRS) able to issuing the damaging management procedure. In fact, there is no support on the EACM for authority disputes since the authority of a management procedure is taken as proven as long as the available security non-repudiation mechanism is correctly used. This mechanism permits that a management server reports any SMF issued by a management client. Optionally, the report can include a resulting configuration status field. This field can be used to indicate the type of damage that resulted to the operational configuration of the server. So, a management server can report problematic management procedures to the Network Management Administration (NMA) using the MDRIS. From this information passed to the MDRSs, the NMA can take punitive actions against the reported management clients. The effectiveness of this system is guaranteed because:
    - Both entities trust the MDRSs of their domain;
    - The mechanism prevents any disputes on the authority of the management procedures since it guarantees the authority of the entity issuing the management procedures;
    - The management procedure result is reported to the NMA through the MDRIS and not directly by any of the entities;
    - A previous analysis of these reports can be made automatically or manually on the MDRSs following some management domain strategy before the NMA is informed; and
    - The MDRSs can send warning messages automatically to the reported management clients before any punitive actions, decided by the NMA, take place.

These mechanisms work together to ensure that the EACM security goals are attainable but some will only work in conjunction with a select group of others. So, each SECM-IDT identifies a standard, pre-defined combination of compatible standard security mechanisms. So, each standard security model is thus a chosen combination of compatible standard security mechanisms. Further more, even the same combination of security mechanisms could generate different security models, depending on the manner the security mechanisms are applied (for example, which key is used in which moment, when to apply the data compression mechanism, etc). Because of this, each security model will have to define the exact methods of application of each security mechanism. Each method will be identified by an additional security mechanism identification tag: an EACM Security Mechanisms Application Method Identification Tag. Until now, the EACM has defined only one method of application of each type of security mechanism, independently of the chosen mechanisms.

4 Evaluations & Conclusions

This document is dedicated to the introduction of the Entity Access Control Model created for and applied to the Network Services Management Framework. This introduction falls short, by evident lack of space, on the detail presented on [1].[i] At this moment, the NSMF, and its integrated EACM, is presented to the computer networks community in general, and to the computer networks management community in particular, as a contribution to overcome some of the most important limitations of present network management frameworks, mainly the Internet Network Management Framework. The creation of the former INSMF was the first effort in this route that was consolidated with the evolution to the NSMF, that can be seen as a manual on *"How to build network services management frameworks"*, applicable to a broad range of network services. One of the larger improvements in the NSMF was the complete definition of the Entity Access Control Model, originally created to be integrated on the INSMF; its definition is complete and independent enough to be applied on other network management frameworks. Moreover, its development was not limited by any commercial or political constrains pursuing only scientific and technological merits. New management frameworks could integrate only some of the concepts and mechanisms defined on the EACM as a way of improvement. New versions of existing frameworks or new frameworks of specific network services could adopt partially or completely this model without having to deploy the entire NSMF framework.

The EACM model can be better evaluated by analyzing its main features:

- The development of this access control model, as with the entire NSMF, was not biased by any commercial or political concerns, often present on the creation and evolution of the major network management frameworks implemented today. Its generality, broad range of supported network management technical features, the universal qualities of the integrated network management concepts and the flexibility of deployment makes the EACM a good candidate for an independent reference on network services management access control models. Furthermore, this model is been presented to be freely scrutinized by the scientific community so it can be corrected and completed, wherever found to be incorrect and incomplete.
- Its definition is based on a set of functional and technical management requisites of modern network services:
    - An architecture of entities organized on hierarchic management domains with identification and addressing capabilities totally independent on the encapsulation transport/network protocols to be used when transferring the management information between management entities.
    - Support for unlimited proxy levels with multiple addressing, including priority address grouping for basic entity backup.
    - An independent distributed service for secure registration, distribution and maintenance of management domains and entities profiles.
    - Support for dynamic entity and network resources control with an unlimited set of types of resources (although using a limited syntax definition), with an unlimited set of resources levels. This control is independent of the identification of the management functions issued and depends on the real resources consumption and the resulting global and specific configuration changes on the target management servers and on the network. This is very important because the management client's access can be controlled based on dynamic results of the management procedures and not on a fixed, predefined type of access to specific objects on management servers. This new approach follows the paradigm of *"controlling of what can be done by the management clients"* and not *"controlling what management client's objects can be read or written"*.
    - Extensive support for security features integrating the most recent security mechanisms and functions. At present, the security models defined as standard on the EACM involve usage if symmetric/private keys but other security paradigms could be defined in the near future since the architecture is totally flexible and can accommodate other types of security flavors without any type of redefinition.
    - Support for local creation of security keys and its encrypted distribution over the network or, if needed, an even securer remote/distributed creation and renovation of security keys without the need to its transfer over the network.

[i] There is an extended version of this article with greater detail on many important aspects of the EACM. Please contact the author for availability.

- Since management domains are organized in administrative or functional hierarchies and resources consumption on management entities and on the network can be controlled with direct resources consumption and complemented with entity proxying, this model is practically immune to scalability problems.
- Almost all features of the EACM started as some technical or functional requisite which resulted on a concept definition and finally on, at least, one standard EACM complete definition using one or several specific mechanisms. The EACM presents all the information needed to these standard mechanisms to be implemented by real network management frameworks. The definitions are made in such a way that the standard mechanisms can be substituted by other mechanisms filling the same requisites and following the same interface syntaxes. This accounts for the EACM great flexibility and upgradeability.

On the other hand, the main limitations of such reference model must be presented so when interest arises on the EACM when planning, defining and/or implementing a network management framework the drawbacks of such approach could be identified as a cost or restrictions of this alternative:

- The EACM most important features have support on the functionality of the Service Management Function concept, so any network management framework adopting this model should integrate a mechanism that mimics this concept introduced by the INSMF/NSMF, noting that only the basic functionalities of the SMFs are needed on the EACM and thus needed to be implemented on such framework.
- At this time, no real implementations of network services management frameworks use the EACM. In fact, there are still no operational code implementations of the EACM alone for real interoperability and functionality tests.
- At the moment, the NSMF and the EACM are not supported by any major international institution/organization dedicated to computer networks management investigation. It is hoped that the former article introducing the INSMF [2], this article and some others in the near future will give, at least, some visibility to this framework and its entity access control model.
- Since the initial inspiration to create the NSMF platform was the INMF, it is likely that the EACM would be accommodated easier on network services management frameworks to be used on the Internet. In fact, the present EACM standard entity addressing definition is to be applicable on the Internet; no other network framework is contemplated with applicable EACM standard address versions at this time.

A quantitative comparison between the NSMF and the INMF was elaborated using a novel network management frameworks evaluation mechanism and presented in [1]. In the specific area of access control and security mechanisms, the EACM was compared against the USM and VACM mechanisms of the INMF and a better aggregate result was achieved: 5.9 against 3.4, respectively (on a scale of 0.0 to 7.0, where higher values are better).

Future Work

The complete and detailed definitions of the NSMF and the EACM are being presented to the scientific community in the last months. This presentation is a mix of limited articles dedicated to the introduction of the most important features of the NSMF or the EACM, a PhD Thesis [1] and a project site hosted at http://gcom.uminho.pt/NSMF. An important step in the future would be the construction of an independent and freely distributed implementation of a network services management framework that followed the EACM complete definition. This could be accomplished by an academic project implemented by a team of software engineers or senior students of one or several universities. Hopefully, these articles and the merits of the NSMF, in general, and of the EACM, in particular, will generate enough interest and support to trigger this kind of projects.

5 Acknowledgments

The author would like to thank Alexandre Santos from the *Departamento de Informática* of the *Universidade do Minho* and Fernando Boavida from the *Departamento de Engenharia Informática* of the *Universidade de Coimbra* for their insight, guidance and overall involvement on the NSMF project.

6 References

[1] B. Dias, *Gestão de Redes Internet – Network Services Management Framework*, PhD Thesis, Universidade do Minho, December 2004.

[2] B. Dias, A. Santos, F. Boavida, *Internet Network Services Management Framework*, Proc. IEEE Int. Conf. on Networks 2002, August 2002.

[3] B.Dias, A. Santos, F. Boavida, *An Introduction to the Network Services Management Framework*, The 3rd IASTED International Conference on Communications, Internet, and Information Technology, November 2004.

[4] J. Case, R. Mundy, D. Partain, B. Stewart, *Introduction and Applicability Statements for Internet Standard Network Management Framework*, Internet Engineering Task Force (IETF), Request for Comments (RFC) 3410, December 2002.

[5] D. Harrington, R. Presuhn, B. Wijnen, *An Architecture for Describing SNMP Management Frameworks*, IETF, RFC 3411, December 2002.

[6] J. Case, D. Harrington, R. Presuhn, B. Wijnen, *Message Processing and Dispatching for the Simple Network Management Protocol, Version 3*, IETF, RFC 3412, December 2002.

[7] D. Levi, P. Meyer, B. Stewart, *SNMP Applications,* IETF, RFC 3413, December 2002.

[8] U. Blumenthal, B. Wijnen, *The User-Based Security Model (USM) for Version 3 of the SNMP*, IETF, RFC 3414, December 2002.

[9] U. Blumenthal, B. Wijnen, *The View-Based Access Control Model (VACM) for Version 3 of the SNMP*, IETF, RFC 3415, December 2002.

[10] K. McCloghrie, R. Presuhn, J. Case, M. Rose, S. Waldbusser, *Version 2 of the Protocol Operations for the Simple Network Management Protocol*, R. Presuhn, Ed., IETF, RFC 3416, December 2002.

[11] K. McCloghrie, R. Presuhn, J. Case, M. Rose, S. Waldbusser, *Transport Mappings for the Simple Network Management Protocol*, R. Presuhn, Ed., IETF, RFC 3417, December 2002.

[12] J. Case, K. McCloghrie, M. Rose, S. Waldbusser, *Management Information Base (MIB) for the Simple Network Management Protocol*, R. Presuhn, Ed., IETF, RFC 3418, December 2002.

[13] S. Waldbusser, *Remote Network Monitoring Management Information Base, Version 2 using SMIv2*, IETF, RFC 2021, January 1997.

[14] D. Levi, J. Schoenwaelder, *Definitions of Managed Objects for Scheduling Management Operations*, IETF, RFC 2591, May 1999.

[15] D. Levi, J. Schoenwaelder, *Definitions of Managed Objects for the Delegation of Management Scripts*, IETF, RFC 3165, August 2001.

[16] D. Levi, J. Schoenwaelder, *Event MIB*, IETF, RFC 2981, August 2000.

[17] B. Stewart, R. Kavasseri - Editor, *Notification Log MIB*, IETF, RFC 3014, November 2000.

[18] B. Stewart, R. Kavasseri - Editor, *Distributed Management Expression MIB*, IETF, RFC 2982, October 2000.

[19] N. Freed, S. Kille, *Network Services Monitoring MIB*, RFC 2788, March 2000.

[20] A. Brites, P. Simões, P. Leitão, E. Monteiro, F. Fernandes, *A High-Level Notation For The Specification Of Network Management Applications*, Proc. INET'94/JENC95.

[21] F. Stamatelopoulos, T. Chiotis, B. Maglaris, *A Scalable, Platform-Based Architecture for Multiple Domain Network Management,* National Technical University of Athens.

[22] G. Goldszmidt, Y. Yemini, *Distributed Management by Delegation*, Proc. 15th Int. Conf. On Distributed Computing Systems, June 1995.

[23] K. Meyer, M. Erlinger, J. Betser, C. Sunshine, *Decentralizing Control and Intelligence in Network Management*, Proc. 4th Int. Symposium on Integrated Network Management, May 1995.

[24] M. Siegl, G. Trausmuth, *Hierarchical Network Management – A concept and its Prototype in SNMPv2*, 1996.

[25] A. Menezes, P.V. Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRCPress, 1996.

[26] H.X. Mel, D. Baker, *Cryptography Decrypted*, Addison-Wesley, 2001.

[27] *Advance Encryption Standard*, Federal Information Processing Standards (FIPS) Pub. 197, National Institutes of Standards and Technology, November 2002.

[28] N. Courtois, J. Pieprzyk, *Cryptanalysis of Block Ciphers with Over-defined Systems of Equations*, Cryptology, November 2002.

[29] E. Filiol*, A New Statistical Testing for Symmetric Ciphers and Hash Functions*, Cryptology, October 2002.

[30] *Data Encryption Standard*, FIPS Pub. 46-3, National Institutes of Standards and Technology, November 2002.

[31] *Triple Data Encryption Algorithm Modes of Operation*, ANSI X9.52-1998, American National Standards Institute. 1998.

[32] *Standard on Computer Data Authentication*, FIPS Pub. 113, National Institutes of Standards and Technology, May 1985.
[33] *The Keyed-Hash Message Authentication Code*, FIPS Pub. 198, National Institutes of Standards and Technology, March 2002.
[34] H. Krawczyk, M. Bellare, R. Canetti, *HMAC: Keyed-Hashing for Message Authentication*, IETF, RFC 2104, February 1997.
[35] R. Rivest, *The MD5 Message-Digest Algorithm,* IETF, RFC 1321, April 1992.
[36] A. Bosselaers, B. den Boer, *Collisions for the compression function of MD5*, Advances in Cryptology, Proc. Eurocrypt'93, LNCS 765, T. Helleseth, Ed., Springer-Verlag, 1994.
[37] A. Bosselaers, B. Preneel, *RIPE – Integrity Primitives for Secure Information Systems. Final Report of RACE, Integrity Primitives Evaluation (RIPE-RACE 1040)*, LNCS 1007, Eds., Springer-Verlag, 1995.
[38] H. Dobbertin, A. Bosselaers, B. Preneel, *RIPEMD-160: A Strengthened Version of RIPEMD, Fast Software Encryption*, LNCS 1039, D. Gollmann, Ed., Springer-Verlag, 1996.
[39] *Secure Hash Standard*, FIPS Pub. 180-1, National Institutes of Standards and Technology, April 1995.