

# Examen RICM 5 – Examens Administration Réseaux

Jean-Luc Richier

22 mars 2012

1 heure – Tous documents autorisés – Répondre sur 1 copie séparée.

Remarque : Calculatrice et tout document autorisés

## 1 Administration de réseau

### 1.1 Question Routage

1. Quels sont les différences entre un routage interne et un routage externe ? Donner des raisons pour choisir l'un ou l'autre type.
2. Un “peering” BGP avec plusieurs autres AS implique-t-il d'accepter du trafic de “transit” ?
3. Qu'est-ce qu'un AS “privé” ? Pourquoi les utilise-t-on ? Peut-il être utilisé si on accepte du trafic de “transit” ?
4. Comment adapter la métrique d'OSPF pour qu'elle reflète la topologie du réseau de manière fidèle ? La configuration par défaut suffit-elle de nos jours ?

### 1.2 Exercice SNMP

Note : On trouvera en annexe les extraits utiles de la MIB-II ; on a supprimé certaines variables pour simplifier, considérez que seules les variables indiquées existent.

1. On considère la variable SysName. Comment peut-on obtenir sa valeur par SNMP ? Donner le nom de la fonction et son ou ses arguments - les “Objet Identifier” (OID) seront donnés sous forme numérique. Quel est le résultat de cet appel ?
2. Peut-on modifier cette variable ? si c'est possible indiquer le nom de la fonction à appeler, son ou ses arguments, et son résultat.
3. On s'intéresse à la lecture de la table des interfaces physiques. Comment récupérer le nom du premier interface existant ? Indiquer le nom de la fonction à appeler, son ou ses arguments, et son résultat.
4. Ecrire un algorithme pour lister les noms des interfaces et leur statut up/down.

### 1.3 Exercice mail

- Sur toutes les machines, l’installation de messagerie a été faite de la manière classique : connectivité par IP, utilisation du transport SMTP, utilisation standard du DNS (i.e. préférer les enregistrements de type MX). On suppose qu’il n’y a aucun problème permanent d’accessibilité.

Note : donner vos réponses en quelques lignes, sans chercher à entrer dans les détails des protocoles.

- Les seuls enregistrements DNS du domaine `essai.fr` sont :

<code>essai.fr.</code>	IN	MX	40	<code>srv2.essai.fr.</code>
<code>essai.fr.</code>	IN	MX	10	<code>srv.essai.fr.</code>
<code>essai.fr.</code>	IN	MX	10	<code>srvmail.essai.fr.</code>
<code>srv.essai.fr.</code>	IN	A	201.1.1.10	
<code>srvmail.essai.fr.</code>	IN	A	201.1.1.1	
<code>srv2.essai.fr.</code>	IN	A	201.1.1.2	

1. La machine `mailserv.test.fr` a un message à envoyer à l’adresse destination **Luc.Dupont@essai.fr** ; que se passe-t-il ?
2. La machine `mailserv.test.fr` a un message à envoyer à l’adresse destination **Luc.Dupont@srv.essai.fr** ; que se passe-t-il ?
3. La machine `mailserv.test.fr` a un message à envoyer à l’adresse destination **Luc.Dupont@admin.essai.fr** ; que se passe-t-il ?
4. Pourquoi y a-t-il 3 enregistrements MX pour `essai.fr` ? Sont-ils tous utilisés en fonctionnement normal ? Y a-t-il des cas où cela peut changer ?

```

RFC1155-SMI DEFINITIONS ::= BEGIN
EXPORTS -- EVERYTHING
    internet, directory, mgmt,
    experimental, private, enterprises,
    OBJECT-TYPE, ObjectName, ObjectSyntax, SimpleSyntax,
    ApplicationSyntax, NetworkAddress, IPAddress,
    Counter, Gauge, TimeTicks, Opaque;

-- the path to the root
internet OBJECT IDENTIFIER ::= { iso(1) org(3) dod(6) 1 }
directory OBJECT IDENTIFIER ::= { internet 1 }
mgmt OBJECT IDENTIFIER ::= { internet 2 }
experimental OBJECT IDENTIFIER ::= { internet 3 }
private OBJECT IDENTIFIER ::= { internet 4 }
enterprises OBJECT IDENTIFIER ::= { private 1 }
-- names of objects in the MIB
ObjectName ::= OBJECT IDENTIFIER

-- syntax of objects in the MIB
ObjectSyntax ::= CHOICE {
    simple SimpleSyntax,
    application-wide ApplicationSyntax
}
SimpleSyntax ::= CHOICE {
    number INTEGER,
    string OCTET STRING,
    object OBJECT IDENTIFIER,
    empty NULL
}
ApplicationSyntax ::= CHOICE {
    address NetworkAddress,
    counter Counter,
    gauge Gauge,
    ticks TimeTicks,
    arbitrary Opaque
}
-- other application-wide types, as they are defined, will be added here
}
-- application-wide types
NetworkAddress ::= CHOICE { internet IPAddress }
IPAddress ::= -- in network-byte order
    [APPLICATION 0] IMPLICIT OCTET STRING (SIZE (4))
Counter ::= [APPLICATION 1] IMPLICIT INTEGER (0..4294967295)
Gauge ::= [APPLICATION 2] IMPLICIT INTEGER (0..4294967295)
TimeTicks ::= [APPLICATION 3] IMPLICIT INTEGER (0..4294967295)
Opaque ::= [APPLICATION 4] -- arbitrary ASN.1 value,
    IMPLICIT OCTET STRING -- "double-wrapped"

```

```

RFC1213-MIB DEFINITIONS ::= BEGIN
IMPORTS mgmt, NetworkAddress, IPAddress, Counter, Gauge, TimeTicks FROM RFC1155-SMI
OBJECT-TYPE FROM RFC-1212;
-- MIB-II (same prefix as MIB-I)
mib-2 OBJECT IDENTIFIER ::= { mgmt 1 }
-- textual conventions
DisplayString ::= OCTET STRING
-- This data type is used to model textual information taken from the NVT ASCII character set.
-- By convention, objects with this syntax are declared as having SIZE (0..255)
.....
-- groups in MIB-II
system OBJECT IDENTIFIER ::= { mib-2 1 }
interfaces OBJECT IDENTIFIER ::= { mib-2 2 }
.....
-- Implementation of the System group is mandatory for all systems.
-- If an agent is not configured to have a value for any of these
-- variables, a string of length 0 is returned.

sysDescr OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION "A textual description of the entity. This value should include
the full name and version identification of the system's hardware
type, software operating-system, and networking software. It is
mandatory that this only contain printable ASCII characters."
 ::= { system 1 }
.....
sysName OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-write
STATUS mandatory
DESCRIPTION "An administratively-assigned name for this managed node.
By convention, this is the node's fully-qualified domain name."
 ::= { system 5 }
.....
-- the Interfaces group
-- Implementation of the Interfaces group is mandatory for all systems.

ifNumber OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "The number of network interfaces (regardless of
their current state) present on this system."
 ::= { interfaces 1 }

```

```

-- the Interfaces table
-- The Interfaces table contains information on the entity's interfaces.
-- Each interface is thought of as being attached to a 'subnetwork'. Note that
-- this term should not be confused with 'subnet' which refers to an addressing
-- partitioning scheme used in the Internet suite of protocols.

ifTable OBJECT-TYPE
    SYNTAX SEQUENCE OF IfEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION "A list of interface entries."
    ::= { interfaces 2 }

ifEntry OBJECT-TYPE
    SYNTAX IfEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION "An interface entry containing objects at the
        subnetwork layer and below for a particular interface."
    INDEX { ifIndex }
    ::= { ifTable 1 }

IfEntry ::=
    SEQUENCE {
        ifIndex INTEGER,                ifDescr DisplayString,
        ifType INTEGER,                ifMtu INTEGER,
        ifSpeed Gauge,                ifPhysAddress PhysAddress,
        ifAdminStatus INTEGER,        ifOperStatus INTEGER,
        ifLastChange TimeTicks,       ifInOctets Counter,
        ifInUcastPkts Counter,        ifInNUcastPkts Counter,
        ifInDiscards Counter,         ifInErrors Counter,
        ifInUnknownProtos Counter,    ifOutOctets Counter,
        ifOutUcastPkts Counter,       ifOutNUcastPkts Counter,
        ifOutDiscards Counter,        ifOutErrors Counter,
        ifOutQLen Gauge,              ifSpecific OBJECT IDENTIFIER
    }

ifIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION "A unique value for each interface. Its value ranges between 1
        and the value of ifNumber. The value for each interface must remain
        constant at least from one re-initialization of the entity's network
        management system to the next re-initialization."
    ::= { ifEntry 1 }

ifDescr OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..255))
    ACCESS read-only
    STATUS mandatory

```

```

DESCRIPTION "A textual string containing information about the interface.
    This string should include the name of the manufacturer, the product
    name and the version of the hardware interface."
    ::= { ifEntry 2 }

ifType OBJECT-TYPE
    SYNTAX INTEGER {
        -- la plupart des valeurs ont etes supprimees
        other(1), -- none of the following
        ethernet-csmacd(6),
        ppp(23),
        softwareLoopback(24),
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION "The type of interface, distinguished according to
        the physical/link protocol(s) immediately 'below'
        the network layer in the protocol stack."
    ::= { ifEntry 3 }
.....
ifAdminStatus OBJECT-TYPE
    SYNTAX INTEGER {
        up(1), -- ready to pass packets
        down(2),
        testing(3) -- in some test mode
    }
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "The desired state of the interface. The testing(3) state
        indicates that no operational packets can be passed."
    ::= { ifEntry 7 }

ifOperStatus OBJECT-TYPE
    SYNTAX INTEGER {
        up(1), -- ready to pass packets
        down(2),
        testing(3) -- in some test mode
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION "The current operational state of the interface. The testing(3)
        state indicates that no operational packets can be passed."
    ::= { ifEntry 8 }
.....
ifInOctets OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION "The total number of octets received on the
        interface, including framing characters."
    ::= { ifEntry 10 }

```