

RICM - troisième année – 2004-2005

Administration de réseaux

Tous documents autorisés – durée indicative 1 heure

Question routage

1. Donner des critères pour décider si deux réseaux sont à mettre dans un même Autonomous System (AS) ou dans deux AS différents.
2. Une entreprise possède deux sites connectés séparément à l'Internet, mais gérés par un même service. Ces deux sites peuvent-ils être dans le même AS ? Commenter.
3. On va ajouter une liaison privée (VPN, ligne, MPLS ...) entre les deux sites, sur laquelle on fera passer le trafic inter-site. Quel type de protocole de routage utiliser sur les routeurs extrémité de la liaison ? Donner une idée des configurations de routage.

Question IPv6

On considère le plan d'attribution d'adresses IPv6 standard (RFC 3587, IPv6 Global Unicast Address Format).

1. Quelle est la longueur d'un préfixe numérotant un câble Ethernet ?
2. Combien de réseaux de niveau 2 (des câbles Ethernet par exemple) peut-on déployer dans un site ?

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. Pour les Objet Identifier (OID), les noter sous forme numérique, complète à partir de la racine.

1. On veut regarder par SNMP si une machine S est un routeur ou non. Quelle variable SNMP consulter ? Donner, le nom, l'OID. Indiquer une requête SNMP pour lire cette variable ; quels sont les arguments de cette requête et le résultat ?
2. Donner l'appel SNMP (avec ses arguments) forçant à 30 le champ « nombre de saut » des paquets IP émis par S.

- On considère pour S la table de routes suivante (cas d'une machine d'interface Ethernet d'adresse 210.1.1.10, netmask 255.255.255.0)

Destination	Routeur	Type	Index-Interface
200.1.1.0/24	210.1.1.1	Gateway	1
default	210.1.1.3	Gateway	1
210.1.1.0/24	210.1.1.10	Direct	1

3. Quelles sont les variables SNMP associées à la route 200.1.1.0/24 (donner les OID et les valeurs) ?
4. Donner un programme pour imprimer cette table de routage. Ne pas chercher à raffiner l'affichage, supposer que vous avez une directive *print* qui accepte tous les types, garder les valeurs numériques.
5. Dans quel ordre votre programme va-t-il imprimer les routes de l'exemple ?

```

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"
END

```

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)

 PhysAddress ::= OCTET STRING

 -- This data type is used to model media addresses. For many types of media, this will be in a binary
 -- representation. For example, an ethernet address would be represented as a string of 6 octets.

-- groups in MIB-II

 system OBJECT IDENTIFIER ::= { mib-2 1 }

 interfaces OBJECT IDENTIFIER ::= { mib-2 2 }

 ip OBJECT IDENTIFIER ::= { mib-2 4 }

.....

-- the IP group

-- Implementation of the IP group is mandatory for all systems.

 ipForwarding OBJECT-TYPE

 SYNTAX INTEGER {

 forwarding(1), -- acting as a gateway

 not-forwarding(2) -- NOT acting as a gateway

 }

 ACCESS read-write

 STATUS mandatory

 DESCRIPTION "The indication of whether this entity is acting as an IP gateway in respect to the forwarding
 of datagrams received by, but not addressed to, this entity. IP gateways forward datagrams.
 IP hosts do not (except those source-routed via the host)."

 ::= { ip 1 }

 ipDefaultTTL OBJECT-TYPE

 SYNTAX INTEGER

 ACCESS read-write

 STATUS mandatory

 DESCRIPTION "The default value inserted into the Time-To-Live field of the IP header of datagrams by the
 transport layer protocol."

 ::= { ip 2 }

 ipInReceives OBJECT-TYPE

 SYNTAX Counter

 ACCESS read-only

 STATUS mandatory

 DESCRIPTION "The total number of input datagrams received from interfaces, including those received in
 error."

 ::= { ip 3 }

.....

-- the IP routing table

-- The IP routing table contains an entry for each route presently known to this entity.

-- NOTE: plusieurs champs ont été supprimés pour simplifier le texte

 ipRouteTable OBJECT-TYPE

 SYNTAX SEQUENCE OF IpRouteEntry

 ACCESS not-accessible

 STATUS mandatory

 DESCRIPTION "This entity's IP Routing table."

 ::= { ip 21 }

```

ipRouteEntry OBJECT-TYPE
    SYNTAX IpRouteEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION "A route to a particular destination."
    INDEX { ipRouteDest }
    ::= { ipRouteTable 1 }

IpRouteEntry ::=
    SEQUENCE {
        ipRouteDest                IpAddress,
        ipRouteIfIndex              INTEGER,
        ipRouteNextHop              IpAddress,
        ipRouteType                  INTEGER,
        ipRouteMask                  IpAddress
    }

ipRouteDest OBJECT-TYPE
    SYNTAX IpAddress
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "The destination IP address of this route.  An entry with a value of 0.0.0.0 is considered a
    default route."
    ::= { ipRouteEntry 1 }

ipRouteIfIndex OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "The index value which uniquely identifies the local interface through which the next hop of
    this route should be reached.  The interface identified by a particular value of this index is
    the one identified by the same value of ifIndex."
    ::= { ipRouteEntry 2 }

ipRouteNextHop OBJECT-TYPE
    SYNTAX IpAddress
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "The IP address of the next hop of this route.  (In the case of a route bound to an interface
    which is realized via a broadcast media, the value of this field is the agent's IP address on
    that interface.)"
    ::= { ipRouteEntry 7 }

ipRouteType OBJECT-TYPE
    SYNTAX INTEGER {
        other(1),                -- none of the following
        invalid(2),              -- an invalidated route
        direct(3),                -- route to directly connected (sub-)network
        indirect(4) }            -- route to a non-local host/network/sub-network
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "The type of route.  Note that the values direct(3) and indirect(4) refer to the notion of direct
    and indirect routing in the IP architecture."
    ::= { ipRouteEntry 8 }

ipRouteMask OBJECT-TYPE
    SYNTAX IpAddress
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "Indicate the mask to be logical-ANDed with the destination address before being compared
    to the value in the ipRouteDest field.
    If the value of the ipRouteDest is 0.0.0.0 (a default route), then the mask value is also
    0.0.0.0.  It should be noted that all IP routing subsystems implicitly use this mechanism."
    ::= { ipRouteEntry 11 }

```

.....

-- Le reste est supprimé