NOMADIX™

XML Interface DTD

Confidential

Release 8.8

30851 Agoura Road, Suite 102
Agoura Hills, CA 91301, USA

++ 1.818.597.1500

http://www.nomadix.com/

NOMADIX™

Contents

Disclaimer

There are a number of caveats that need to be expressly stated:

1. Nomadix does not guarantee that following these guidelines will ensure the problem-free interoperability between the web server running the XML scripts and Nomadix technology.

2. To ensure accuracy for future releases, Nomadix reserves the right to change and add to this specification without notice.

1. XML Command Authentication

All XML commands are authenticated before being acted upon. The NSE supports two methods of authentication, each of which can be enabled/disabled via configuration.

- Authenticate via IP address – This legacy method of authentication verifies that the IP address of the peer (the sender of the XML command) matches one of the configured peer addresses (referred to as XML Server 1 through XML Server 4)
- Authenticate via user credentials – This method of authentication verifies the user credentials supplied in the HTTP request used to transport the XML command. Commands that update the NSE's operating state require a privilege level of "administrator" or "XML". The "operator" privilege level is also allowed for commands that simply retrieve state.

All XML commands are available via the HTTP port (80) or the HTTPS port (443). That is, commands that specify a port of 1111 or 1112 are also available on port 80 or 443.

Commands that specify a port of 80 or 443 are only available on those ports (i.e. they are not available on ports 1111 or 1112).

Both authentication methods described above are available on the HTTP port (80) and the HTTPS port (443).

Only the first authentication method (Authenticate via IP address) is available on port 1111 and port 1112.

2. Radius Subscriber Administration Commands

NOTE: The commands listed in this section should be sent as a POST to one of the following addresses:

http://NSE_URI:1111/usg/command.xml
https://NSE_URI:1112/usg/command.xml

Please note the port difference between standard and secure transmissions.

2.1 User Login Command for Radius Subscriber Login

The Portal Page web server can send this command to instruct the NSE to send a RADIUS authentication request to the RADIUS server to authenticate a subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Login command sent to NSE
-->

<!ELEMENT SUB_USER_NAME (#PCDATA)>
<!ELEMENT SUB_PASSWORD (#PCDATA)>
<!ELEMENT SUB_MAC_ADDR (#PCDATA)>
<!ELEMENT PORTAL_SUB_ID (#PCDATA)>

<!ELEMENT USG (SUB_USER_NAME, SUB_PASSWORD, SUB_MAC_ADDR,
PORTAL_SUB_ID?)>
<!ATTLIST USG COMMAND CDATA #REQUIRED>
```

Where:
COMMAND attribute: 'RADIUS_LOGIN'
SUB_USER_NAME: Subscriber's username (char [96])
SUB_PASSWORD: Subscriber's password (char [128])
SUB_MAC_ADDR: Subscriber's MAC address (char [12])
PORTAL_SUB_ID (optional): Unique identifier that the Portal Page web server can send to the NSE which will be sent back with status response (char [36])

Sample command XML:

```
<USG COMMAND="RADIUS_LOGIN">
    <SUB_USER_NAME>jsmith</SUB_USER_NAME>
    <SUB_PASSWORD>abc123</SUB_PASSWORD>
    <SUB_MAC_ADDR>1A2B3C4D5E6F</SUB_MAC_ADDR>
    <PORTAL_SUB_ID>12345678-1234-1234-1234-123456789012</PORTAL_SUB_ID>
</USG>
```

Response for the Login Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

The NSE will send status message asynchronously if the "Portal XML POST URL" is enabled in the AAA section of the NSE (see User Status Message section).

2.2 User Logout Command for Radius Subscriber Logout

The Portal Page web server can send this command to instruct the NSE to logout the subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Logout command sent to NSE
-->

<!ELEMENT SUB_MAC_ADDR (#PCDATA)>
<!ELEMENT SUB_USER_NAME (#PCDATA)>

<!ELEMENT USG (SUB_MAC_ADDR, SUB_USER_NAME)>
<!ATTLIST USG COMMAND CDATA #REQUIRED>
```

Where:
COMMAND attribute: 'LOGOUT'
SUB_MAC_ADDR: Subscriber's MAC address (char [12], optional if username is present)
SUB_USER_NAME: Subscriber's username (char [96], optional if MAC address is present)

Sample command XML:

```
<USG COMMAND="LOGOUT">
    <SUB_MAC_ADDR>1A2B3C4D5E6F</SUB_MAC_ADDR>
    <SUB_USER_NAME>jsmith</SUB_USER_NAME>
</USG>
```

Response for the Logout Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

The NSE will send status message asynchronously if the "Portal XML POST URL" is enabled in the AAA section of the NSE (see User Status Message).

3. Subscriber Administration Commands

NOTE: *Unless specified otherwise,* the commands listed in this section should be sent as a POST to one of the following addresses:

http://NSE_URI:1111/usg/command.xml
https://NSE_URI:1112/usg/command.xml

Please note the port difference between standard and secure transmissions.

3.1 User Add Command

The specified subscriber has been authorized for access and will be added to the NSE's MAC authorization table. If the subscriber is in the 'Current' (active) memory table of the NSE then the Update Cache XML command must follow in order to correctly update the subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Add command sent to NSE
-->

<!ELEMENT USER_NAME (#CDATA)>
<!ELEMENT PASSWORD(#PCDATA)>
<!ELEMENT EXPIRY_TIME (#PCDATA)>
<!ELEMENT COUNTDOWN (#PCDATA)>
<!ELEMENT ROOM_NUMBER (#PCDATA)>
<!ELEMENT PAYMENT_METHOD (#PCDATA)>
<!ELEMENT PLAN (#PCDATA)>
<!ELEMENT IP_TYPE (#PCDATA)>
<!ELEMENT DHCP_SUBNET(#PCDATA)>
<!ELEMENT CONFIRMATION (#PCDATA)>
<!ELEMENT PAYMENT (#PCDATA)>
<!ELEMENT USER_DEF1 (#CDATA)>
<!ELEMENT USER_DEF2 (#CDATA)>
<!ELEMENT SMTP_REDIRECT (#PCDATA)>
<!ELEMENT BANDWIDTH_UP (#PCDATA)>
<!ELEMENT BANDWIDTH_DOWN (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_UP (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_DOWN (#PCDATA)>
<!ELEMENT QOS_POLICY (#PCDATA)>
<!ELEMENT CLASS_NAME (#PCDATA)>

<!ELEMENT USG (USER_NAME?, PASSWORD?, EXPIRY_TIME?, COUNTDOWN?,
ROOM_NUMBER?, PAYMENT_METHOD, PLAN?, IP_TYPE?, DHCP_SUBNET?,
CONFIRMATION?, PAYMENT?, USER_DEF1?, USER_DEF2?, SMTP_REDIRECT?,
BANDWIDTH_UP?, BANDWIDTH_DOWN?, BANDWIDTH_MAX_UP?,
BANDWIDTH_MAX_DOWN?, QOS_POLICY?,CLASS_NAME?)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
MAC_ADDR CDATA
>
<!ATTLIST PASSWORD ENCRYPT (TRUE | FALSE) #REQUIRED >
<!ATTLIST EXPIRY_TIME        UNITS (SECONDS | MINUTES | HOURS | DAYS) #REQUIRED >
```

Where:
COMMAND attribute: USER_ADD
MAC_ADDR attribute (optional): Subscriber's MAC address (char [12])
USER_NAME (optional): Subscriber's username (char [96])
PASSWORD (optional): Subscriber's password (char [128])
ENCRYPT attribute: Either TRUE or FALSE
EXPIRY_TIME (optional): Expiry time
UNITS attribute: Either SECONDS, MINUTES, HOURS or DAYS
ROOM_NUMBER (optional): (char [8])

PAYMENT_METHOD (optional but recommended): Either "RADIUS", "PMS", "CREDIT_CARD", or "ROOM_OPEN"
IP_TYPE (optional): Either “PRIVATE” or “PUBLIC”
DHCP_SUBNET (optional): Subnet based on configured DHCP subnets in the NSE
CONFIRMATION (optional): Confirmation number/ID
PAYMENT (optional): Amount charged for access
USER_DEF1 (optional): User definable string (char [128]), if not provided in the command, NSE will empty it.
USER_DEF2 (optional): User definable string (char [128]), if not provided in the command, NSE will empty it.
COUNTDOWN (optional): 0 off, 1 enabled. If not present, it defaults to off. (NOTE: If a billing plan is specified and it is an X-over-Y billing plan, then the *countdown* element, if present, is irrelevant and is ignored.
PLAN: (optional): This relates to the X over Y plan number in Billing Plans setup. If used for X over Y, USER_NAME and PASSWORD are required.
SMTP_REDIRECT: (optional): Either TRUE or FALSE for SMTP Redirection enabled for that user. If not included, the User will have this variable as TRUE for their profile.
BANDWIDTH_UP: (optional): This will set the Upstream Bandwidth for a user without having to send the other Bandwidth XML command. Legacy element that is obsolete because of Bandwidth_Max_Up.
BANDWIDTH_DOWN: (optional): This will set the Downstream Bandwidth for a user without having to send the other Bandwidth XML Command. Legacy element that is obsolete because of Bandwidth_Max_Down.
BANDWIDTH_MAX_UP: (optional): This will set the Maximum Upstream bandwidth for the user without having to send the other Bandwidth XML Command.
BANDWIDTH_MAX_DOWN: (optional): This will set the Maximum Downstream bandwidth for the user without having to send the other Bandwidth XML Command.
QOS_POLICY: (optional): Select and add the QoS Policy that is configured on the NSE to the profile for the user.
CLASS_NAME: (optional): Class name (char [64]) indicates the class that traffic to/from this user should be assigned to for Class-Based Queuing purposes.

Sample command XML (Normal Plan):

```
<USG COMMAND="USER_ADD" MAC_ADDR="1A2B3C4D5E6F">
    <USER_NAME><![CDATA[jsmith]]></USER_NAME>
    <PASSWORD ENCRYPT="FALSE">JSMITH6</PASSWORD>
    <EXPIRY_TIME UNITS="SECONDS">60</EXPIRY_TIME>
    <COUNTDOWN>1</COUNTDOWN>
    <ROOM_NUMBER>1234</ROOM_NUMBER>
    <PAYMENT_METHOD>CREDIT_CARD</PAYMENT_METHOD>
    <IP_TYPE>PRIVATE</IP_TYPE>
    <DHCP_SUBNET>192.168.1.0</DHCP_SUBNET>
    <CONFIRMATION>123abc</CONFIRMATION>
    <PAYMENT>9.95</PAYMENT>
    <USER_DEF1><![CDATA[meeting room]]></USER_DEF1>
    <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
    <SMTP_REDIRECT>TRUE</SMTP_REDIRECT>
    <BANDWIDTH_MAX_UP>256</BANDWIDTH_MAX_UP>
    <BANDWIDTH_MAX_DOWN>256</BANDWIDTH_MAX_DOWN>
    <QOS_POLICY>QoSPolicy1</QOS_POLICY>
    < CLASS_NAME> Lobby</CLASS_NAME>
</USG>
```

Sample command XML (X over Y Plan):

```
<USG COMMAND="USER_ADD" MAC_ADDR="1A2B3C4D5E6F">
    <USER_NAME><![CDATA[jsmith]]></USER_NAME>
    <PASSWORD ENCRYPT="FALSE">JSMITH6</PASSWORD>
    <PAYMENT_METHOD>CREDIT_CARD</PAYMENT_METHOD>
    <PLAN>0</PLAN>
    <USER_DEF1><![CDATA[meeting room]]></USER_DEF1>
    <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
    <SMTP_REDIRECT>TRUE</SMTP_REDIRECT>
    <BANDWIDTH_MAX_UP>256</BANDWIDTH_MAX_UP>
    <BANDWIDTH_MAX_DOWN>256</BANDWIDTH_MAX_DOWN>
    <QOS_POLICY>QoSPolicy1</QOS_POLICY>
</USG>
```

Response for the User Add Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.2 Device Add Command

In which a specified Device is authorized for access, and is added to the NSE authorized MAC address database. The device is furthermore *guaranteed* access at any time by reserving a permanent entry for it in the NSE Current (active) subscriber table.

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Device Add command sent to NSE
-->

<!ELEMENT DEVICE_NAME (#CDATA)>
<!ELEMENT IP_ADDR (#PCDATA)>
<!ELEMENT DHCP_SUBNET(#PCDATA)>
<!ELEMENT PROXY_ARP (#PCDATA)>
<!ELEMENT VLAN (#PCDATA)>
<!ELEMENT USER_DEF1 (#CDATA)>
<!ELEMENT USER_DEF2 (#CDATA)>
<!ELEMENT SMTP_REDIRECT (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_UP (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_DOWN (#PCDATA)>
<!ELEMENT QOS_POLICY (#PCDATA)>
<!ELEMENT CLASS_NAME (#PCDATA)>

<!ELEMENT USG (DEVICE_NAME?,
                        IP_ADDR?, DHCP_SUBNET?,
                        PROXY_ARP?, VLAN?,
                        USER_DEF1?, USER_DEF1?,
                        SMTP_REDIRECT?,
                        BANDWIDTH_MAX_UP?, BANDWIDTH_MAX_DOWN?,
                        QOS_POLICY?, CLASS_NAME?)>

<!ATTLIST USG COMMAND  CDATA #REQUIRED
                      MAC_ADDR CDATA #REQUIRED>
```

Where:

Attributes (mandatory)

COMMAND	DEVICE_ADD
MAC_ADDR	MAC address of the device Should a prior Device ***or User entry*** exists in the NSE database for this same MAC address, it will be overwritten by this command.

Elements (optional)

DEVICE_NAME	A short **name** for the device (char[96]) to assist administrator or operator recognition of it.
IP_ADDR	The **IP address** associated with the device, if any.
DHCP_SUBNET	The **IP subnet mask** of an NSE **DHCP address pool** that the device shall draw its IP address from, should it need to.
PROXY_ARP	Enable (TRUE) or disable (FALSE) **Proxied ARP** for this device. Proxied ARP allows a device to communicate directly with other subscriber-side devices, similarly configured, regardless of which VLAN those devices are attached to.
VLAN	**802.1Q VLAN port** that device is attached to ($0 \leq$ VLAN $\leq$ 4095). If omitted or zero, the device will be granted access no matter where it has attached; but if a non-zero VLAN is specified, the device will only be granted access when attached to that VLAN.
USER_DEF1	**User-defined content** associated with the device, comprising any sequence of UTF-8 code points that does not exceed 128 *octets* in length.
USER_DEF2	**User-defined content** associated with the device, comprising any sequence of UTF-8 code points that does not exceed 128 *octets* in length.
SMTP_REDIRECT	Enable (TRUE) or disable (FALSE) **SMTP Redirection** for the device.
BANDWIDTH_MAX_UP	**Maximum Upstream Bandwidth** to be granted the device.
BANDWIDTH_MAX_DOWN	**Maximum Downstream Bandwidth** to be granted this device.
QOS_POLICY	The name of any NSE **QoS Policy** that should apply to the device.
CLASS_NAME	The name of any NSE **Bandwidth Class** that should apply to the device.

Sample command XML:

```
<USG COMMAND="DEVICE_ADD" MAC_ADDR="1A2B3C4D5E6F">
    <DEVICE_NAME><![CDATA[Erasmus]]></USER_NAME>
    <IP_ADDR>192.168.1.44</IP_ADDR>
    <DHCP_SUBNET>192.168.1.0</DHCP_SUBNET>
    <PROXY_ARP>TRUE</PROXY_ARP>
    <VLAN>14</VLAN>
    <USER_DEF1><![CDATA[meeting room]]></USER_DEF1>
    <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
    <SMTP_REDIRECT>TRUE</SMTP_REDIRECT>
    <BANDWIDTH_MAX_UP>256</BANDWIDTH_MAX_UP>
    <BANDWIDTH_MAX_DOWN>256</BANDWIDTH_MAX_DOWN>
    <QOS_POLICY>QoSPolicy1</QOS_POLICY>
    < CLASS_NAME>Lobby</CLASS_NAME>
</USG>
```

Response for the Device Add Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see “Standard OK/ERROR Response” section for DTD definition).

3.3 Group Add Command

The Specified Group is added to the authorized database of the NSE and utilizes the listed attributes for the group. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Group Add command sent to NSE
-->

<!ELEMENT USER_NAME (#CDATA)>
<!ELEMENT PASSWORD(#PCDATA)>
<!ELEMENT EXPIRY_TIME (#PCDATA)>
<!ELEMENT DHCP_TYPE (#PCDATA)>
<!ELEMENT DHCP_SUBNET(#PCDATA)>
<!ELEMENT PAYMENT (#PCDATA)>
<!ELEMENT USER_DEF1 (#CDATA)>
<!ELEMENT USER_DEF2 (#CDATA)>
<!ELEMENT SMTP_REDIRECT (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_UP (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_DOWN (#PCDATA)>
<!ELEMENT QOS_POLICY (#PCDATA)>
<!ELEMENT GROUP_USERS_MAX (#PCDATA)>
<!ELEMENT CLASS_NAME (#PCDATA)>
<!ELEMENT VALID_UNTIL  (#PCDATA)>

<!ELEMENT USG (USER_NAME, PASSWORD, EXPIRY_TIME, DHCP_TYPE?, DHCP_SUBNET?,
PAYMENT?, USER_DEF1?, USER_DEF1?, SMTP_REDIRECT?, BANDWIDTH_MAX_UP?,
BANDWIDTH_MAX_DOWN?, QOS_POLICY?, GROUP_USERS_MAX?,CLASS_NAME?,
VALID_UNTIL?)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
>
<!ATTLIST PASSWORD ENCRYPT (TRUE | FALSE) #REQUIRED >
<!ATTLIST EXPIRY_TIME          UNITS (SECONDS | MINUTES | HOURS | DAYS) #REQUIRED >
```

Where:
COMMAND attribute: GROUP_ADD
USER_NAME (Required): Group's username (char [96])
PASSWORD (Required): Group's password (char [128])
ENCRYPT attribute: Either TRUE or FALSE
EXPIRY_TIME (Required): Expiry time
UNITS attribute: Either SECONDS, MINUTES, HOURS or DAYS
DHCP_TYPE (optional): Either "PRIVATE" or "PUBLIC"
DHCP_SUBNET (optional): Subnet based on configured DHCP subnets in the NSE
PAYMENT (optional): Amount charged for access

USER_DEF1 (optional): User definable string (char [128]), if not provided in the command, NSE will empty it.
USER_DEF2 (optional): User definable string (char [128]), if not provided in the command, NSE will empty it.
SMTP_REDIRECT: (optional): Either TRUE or FALSE for SMTP Redirection enabled for that user. If not included the User will have this variable as TRUE for their profile.
BANDWIDTH_MAX_UP: (optional): This will set the Maximum Upstream bandwidth for the user without having to send the other Bandwidth XML Command.
BANDWIDTH_MAX_DOWN: (optional): This will set the Maximum Downstream bandwidth for the user without having to send the other Bandwidth XML Command.
QOS_POLICY (optional): Select and add the QoS Policy that is configured on the NSE to the profile for the user.
GROUP_USERS_MAX (optional): This will set the maximum number of concurrent users that can utilize this Group account.
CLASS_NAME: (optional): Class name (char [64]) indicates the class that traffic to/from this user should be assigned to for Class-Based Queuing purposes.
VALID_UNTIL: (optional): The date/time at which this group will cease to exist. If non-empty, must be expressed in a valid ISO 8601 format. Absence of this element or an empty string means the group will have permanent (until administratively deleted) existence. A date/time that does not lie in the future (with respect to the NSE's current time) will be rejected as an error. The granularity of this parameter is in minutes, so if the ISO 8601 string includes seconds they will be ignored (i.e., treated as if submitted as "00").

Sample command XML:

```
<USG COMMAND="GROUP_ADD">
    <USER_NAME><![CDATA[Conference1]]></USER_NAME>
    <PASSWORD ENCRYPT="FALSE">users</PASSWORD>
    <EXPIRY_TIME UNITS="SECONDS">600</EXPIRY_TIME>
    <DHCP_SUBNET>192.168.1.0</DHCP_SUBNET>
    <DHCP_TYPE>PRIVATE</DHCP_TYPE>
    <PAYMENT>9.95</PAYMENT>
    <USER_DEF1><![CDATA[meeting room1]]></USER_DEF1>
    <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
    <SMTP_REDIRECT>TRUE</SMTP_REDIRECT>
    <BANDWIDTH_MAX_UP>256</BANDWIDTH_MAX_UP>
    <BANDWIDTH_MAX_DOWN>256</BANDWIDTH_MAX_DOWN>
    <QOS_POLICY>QoSPolicy1</QOS_POLICY>
    <GROUP_USERS_MAX>25</GROUP_USERS_MAX>
    < CLASS_NAME> Lobby</CLASS_NAME>
    < VALID_UNTIL>2014-08-15T11:00-07:00</VALID_UNTIL>
</USG>
```

Response for the Group Add Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.4 Update Cache Command

The memory authorization table entry specified by the MAC address will have its status changed from "pending" to "authorized". NOTE: It is important to update the cache to enable proper access for the subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Update Cache command sent to NSE
-->

<!ELEMENT PAYMENT_METHOD (#PCDATA)>

<!ELEMENT USG (PAYMENT_METHOD?)>
<!ATTLIST USG
COMMAND CDATA #REQUIRED
MAC_ADDR CDATA #REQUIRED
>
```

Where:
COMMAND attribute: CACHE_UPDATE
MAC_ADDR attribute: Subscriber's MAC address (char [12])

Sample command XML:

```
<USG COMMAND="CACHE_UPDATE" MAC_ADDR="1A2B3C4D5E6F">
</USG>
```

Response for the Update Cache Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.5 Bandwidth Up Command

Set the Bandwidth Up for an authorized subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Bandwidth Up command sent to NSE
-->

<!ELEMENT BANDWIDTH_UP (#PCDATA)>

<!ELEMENT USG (BANDWIDTH_UP)>
<!ATTLIST USG
COMMAND CDATA #REQUIRED
        SUBSCRIBER CDATA #REQUIRED
>
```

Where:
COMMAND attribute: SET_BANDWIDTH_UP
SUBSCRIBER attribute: Subscriber's MAC address (char [12])
BANDWIDTH_UP: (number measured in Kbps (i.e. for 128,000 bits per second, enter 128))

Sample command XML:

```
<USG COMMAND="BANDWIDTH_UP" SUBSCRIBER="1A2B3C4D5E6F">
    <BANDWIDTH_UP>128</BANDWIDTH_UP>
</USG>
```

Response for the Bandwidth Up Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.6 Bandwidth Down Command

Set the Bandwidth Down for an authorized subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Bandwidth Down command sent to NSE
-->

<!ELEMENT BANDWIDTH_DOWN (#PCDATA)>

<!ELEMENT USG (BANDWIDTH_DOWN)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
SUBSCRIBER CDATA #REQUIRED
>
```

Where:
COMMAND attribute: SET_BANDWIDTH_DOWN
SUBSCRIBER attribute: Subscriber's MAC address (char [12])
BANDWIDTH_DOWN: (number measured in Kbps (i.e. for 128,000 bits per second, enter 128))

Sample command XML:

```
<USG COMMAND="BANDWIDTH_DOWN" SUBSCRIBER="1A2B3C4D5E6F">
    <BANDWIDTH_DOWN>256</BANDWIDTH_DOWN>
</USG>
```

Response for the Bandwidth Down Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.7 Max Bandwidth Down Command

Set the guaranteed Maximum Downstream Bandwidth for an Authorized Subscriber.. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Bandwidth Max Down command sent to NSE
-->

<!ELEMENT BANDWIDTH_MAX_DOWN (#PCDATA)>

<!ELEMENT USG (BANDWIDTH_MAX_DOWN)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
SUBSCRIBER CDATA #REQUIRED
>
```

Where:
COMMAND attribute: SET_BANDWIDTH_MAX_DOWN
SUBSCRIBER attribute: Subscriber's MAC address (char [12])
BANDWIDTH_MAX_DOWN: (number measured in Kbps (i.e. for 128,000 bits per second, enter 128))

Sample command XML:

```
<USG COMMAND="SET_BANDWIDTH_MAX_DOWN" SUBSCRIBER="1A2B3C4D5E6F">
    <BANDWIDTH_MAX_DOWN>256</BANDWIDTH_MAX_DOWN>
</USG>
```

Response for the Bandwidth Max Down Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.8 Max Bandwidth Up Command

Set the guaranteed Maximum Upstream Bandwidth for an Authorized Subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Bandwidth Max Up command sent to NSE
-->

<!ELEMENT BANDWIDTH__MAX_UP (#PCDATA)>

<!ELEMENT USG (BANDWIDTH_MAX_UP)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
SUBSCRIBER CDATA #REQUIRED
>
```

Where:
COMMAND attribute: SET_BANDWIDTH_MAX_UP
SUBSCRIBER attribute: Subscriber's MAC address (char [12])
BANDWIDTH_MAX_UP: (number measured in Kbps (i.e. for 128,000 bits per second, enter 128))

Sample command XML:

```
<USG COMMAND="SET_BANDWIDTH_MAX_UP" SUBSCRIBER="1A2B3C4D5E6F">
    <BANDWIDTH_MAX_UP>256</BANDWIDTH__MAX_UP>
</USG>
```

Response for the Bandwidth Max Up Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.9 User Payment Command

Subscriber's authorization and payment is requested. The authorization method can only be set to PMS. The NSE will verify room mapping, establish communication with the PMS system, post access fee to the PMS for the subscriber's room bill and add the subscriber to the internal database for access. If the subscriber is in the Current (active) memory table of the NSE then the Update Cache XML command must follow in order to correctly update the subscriber. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  DTD defines User Payment command sent to NSE
-->
<!ELEMENT USER_NAME (#PCDATA)>
<!ELEMENT REAL_NAME (#PCDATA)>
<!ELEMENT PASSWORD (#PCDATA)>
<!ELEMENT EXPIRY_TIME (#PCDATA)>
<!ELEMENT ROOM_NUMBER (#PCDATA)>
<!ELEMENT PAYMENT (#PCDATA)>
<!ELEMENT MAC_ADDR (#PCDATA)>
<!ELEMENT REG_NUMBER (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_UP (#PCDATA)>
<!ELEMENT BANDWIDTH_MAX_DOWN (#PCDATA)>
<!ELEMENT COUNTDOWN (#PCDATA)>
<!ELEMENT BILLING_PLAN (#PCDATA)>
<!ELEMENT CC_SUFFIX (#PCDATA)>
<!ELEMENT CC_EXPIRATION (#PCDATA)>
<!ELEMENT WFB_BUNDLED (#PCDATA)>
<!ELEMENT TRANS_ID (#PCDATA)>
<!ELEMENT REVENUE_CENTER (#PCDATA)>
<!ELEMENT CLASS_NAME (#PCDATA)>

<!ELEMENT USG (USER_NAME, REAL_NAME?, PASSWORD, EXPIRY_TIME, ROOM_NUMBER,
PAYMENT?, MAC_ADDR?, REG_NUMBER?, BANDWIDTH_MAX_UP?,
BANDWIDTH_MAX_DOWN?, COUNTDOWN?, BILLING_PLAN?, CC_SUFFIX?,
CC_EXPIRATION?, WFB_BUNDLED?, TRANS_ID?, REVENUE_CENTER?,CLASS_NAME?)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
        PAYMENT_METHOD CDATA #REQUIRED
>
<!ATTLIST PASSWORD ENCRYPT (TRUE | FALSE) #REQUIRED>
<!ATTLIST EXPIRY_TIME UNITS (SECONDS | MINUTES | HOURS | DAYS) #REQUIRED>
<!ATTLIST WFB_BUNDLED WFB_OPTION (A | B | C | D) #IMPLIED>
```

Where:
COMMAND attribute: USER_PAYMENT
PAYMENT_METHOD attribute: 'PMS'

USER_NAME: Subscriber's username (char [96]). Note: For 2-way PMS, the subscriber's MAC address is optional but recommended.
REAL_NAME (optional, but required for 2-way PMS): Subscriber's real name as listed in PMS (char [96])
PASSWORD: Subscriber's password (char [128])
ENCRYPT attribute: Either TRUE or FALSE
EXPIRY_TIME (optional, but required for 2-way PMS): Expiry time
UNITS attribute: Either SECONDS, MINUTES, HOURS or DAYS
ROOM_NUMBER: Room number (Port-Location "Location" number) of access (char [8]). Note: For 2-way PMS, use the PMS database room number.
PAYMENT (optional): Amount charged for access
MAC_ADDR: MAC address of user for post-paid PMS and 2-way PMS (char [12]).
REG_NUMBER: Reservation number of hotel guest for Micros Fidelio FIAS compliant Query and Post interface (char [24]).
BANDWIDTH_MAX_UP: (optional): This will set the Maximum Upstream bandwidth for the user without having to send any other Bandwidth XML Command.
BANDWIDTH_MAX_DOWN: (optional): This will set the Maximum Downstream bandwidth for the user without having to send any other Bandwidth XML Command.
COUNTDOWN: (optional): This will set the user so that their allotted time will not start counting down, and the charge will not post, until they log in (note: only supported for 1-way PMS systems).
BILLING_PLAN: (optional): This will allow selection of a specified billing plan for either an X over Y Setting or a WFB selection for the user.
CC_SUFFIX: (optional): Last 4 Digits of the Credit Card for Marriott WFB PMS Verification.
CC_EXPIRATION: (optional): Expiration Date on the Credit Card for Marriott WFB PMS Verification. Format = MMYY.
WFB_BUNDLED: (optional): WFB Bundle Bill. 0 = Charge 1 = Bundle
WFB_OPTION attribute: Either A, B, C or D
TRANS_ID: (optional): (32 bit unsigned Integer) Used to match commands with USER_STATUS messages. Information entered here will be mirrored on the USER_STATUS messages.
REVENUE_CENTER: (optional): 3 Digits to specify the Revenue Center for MICROS PMS, or 2 Digits to specify Revenue Code for Marriott WFB and Marriott FOSSE.
CLASS_NAME: (optional): Class name (char [64]) indicates the class that traffic to/from this subscriber should be assigned to for Class-Based Queuing purposes.

Sample command XML (Micros Fidelio FIAS Query and Post):

```
<USG COMMAND="USER_PAYMENT" PAYMENT_METHOD="PMS">
    <USER_NAME>jsmith</USER_NAME>
    <REAL_NAME></REAL_NAME>
    <PASSWORD ENCRYPT="FALSE">JSMITH</PASSWORD>
    <EXPIRY_TIME UNITS="SECONDS">60</EXPIRY_TIME>
    <ROOM_NUMBER>1234</ROOM_NUMBER>
    <PAYMENT>9.95</PAYMENT>
    <MAC_ADDR>001122334455</MAC_ADDR>
    <REG_NUMBER>0123456789</REG_NUMBER>
    <BANDWIDTH_MAX_UP>256</BANDWIDTH_MAX_UP>
    <BANDWIDTH_MAX_DOWN>256</BANDWIDTH_MAX_DOWN>
    <CLASS_NAME>Lobby</CLASS_NAME>
</USG>
```

Sample command XML (2-Way PMS):

```
<USG COMMAND="USER_PAYMENT" PAYMENT_METHOD="PMS">
    <USER_NAME>001122334455</USER_NAME>
    <REAL_NAME>Smith</REAL_NAME>
    <PASSWORD ENCRYPT="FALSE">JSMITH</PASSWORD>
    <EXPIRY_TIME UNITS="SECONDS">3600</EXPIRY_TIME>
    <ROOM_NUMBER>1234</ROOM_NUMBER>
    <PAYMENT>9.95</PAYMENT>
    <MAC_ADDR>0010a4a9cc19</MAC_ADDR>
    <BANDWIDTH_MAX_UP>256</BANDWIDTH_MAX_UP>
    <TRANS_ID>123546</TRANS_ID>
    <WFB_BUNDLED>0</WFB_BUNDLED>
    <COUNTDOWN>DISABLED</COUNTDOWN>
    <CLASS_NAME>Lobby</CLASS_NAME>
</USG>
```

Response for the User Payment Command

This is the response sent to User Payment command. The response is an XML message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for User Payment command
-->

<!ELEMENT CONFIRMATION (#PCDATA)>

<!ELEMENT USG (CONFIRMATION)>

<!ATTLIST USG
RESULT CDATA #REQUIRED
            ID CDATA #REQUIRED
            IP CDATA #REQUIRED
>
```

Where:
CONFIRMATION: confirmation number/ID
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])

Sample Response XML:

```
<USG RESULT="OK" ID="ABC1234" IP="192.168.100.102">
    <CONFIRMATION>123abc</CONFIRMATION>
</USG>
```

3.10 User Delete Command

The subscriber's specified by MAC address or username, will be deleted from the authorization table. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Delete command sent to NSE
-->

<!ELEMENT USER (#PCDATA)>

<!ELEMENT USG (USER)>

<!ATTLIST USG COMMAND CDATA #REQUIRED>
<!ATTLIST USER ID_TYPE (MAC_ADDR |  USER_NAME) #REQUIRED>
```

Where:
COMMAND attribute: USER_DELETE
USER attribute: ID_TYPE (either MAC_ADDR or USER_NAME)
MAC_ADDR: Subscriber's MAC address (char [12], optional if username is present)
USER_NAME: Subscriber's username (char [96], optional if MAC is present)

Sample command XML:

```
<USG COMMAND="USER_DELETE">
    <USER ID_TYPE="MAC_ADDR">001122334455</USER>
</USG>
```

or

```
<USG COMMAND="USER_DELETE">
    <USER ID_TYPE="USER_NAME">jsmith</USER>
</USG>
```

Response for the User Delete Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.11 Device Delete Command

The device specified by MAC address will be deleted from the authorization table. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Device Delete command sent to NSE
-->

<!ELEMENT MAC_ADDR (#PCDATA)>

<!ELEMENT USG (MAC_ADDR)>

<!ATTLIST USG COMMAND  CDATA #REQUIRED
                       MAC_ADDR CDATA #REQUIRED>
```

Where:

Attributes (mandatory)

COMMAND	DEVICE_DELETE
MAC_ADDR	MAC address of the device

Sample command XML:

```
<USG COMMAND="DEVICE_DELETE" MAC_ADDR="001122334455">
</USG>
```

Response for the Device Delete Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see "Standard OK/ERROR Response" section for DTD definition).

3.12 User Query Command

The user's data contained in the authorization table is returned (a listing for the user being queried must be present in the Current table for the command to complete successfully). This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Query command sent to NSE
-->

<!ELEMENT USER (#PCDATA)>
<!ELEMENT USG (USER)>

<!ATTLIST USG COMMAND CDATA #REQUIRED>
<!ATTLIST USER ID_TYPE (MAC_ADDR | USER_NAME) #REQUIRED>
```

Where:
COMMAND attribute: USER_QUERY
USER attribute: ID_TYPE (either MAC_ADDR or USER_NAME
MAC_ADDR: Subscriber's MAC address (char [12], optional if username is present)
USER_NAME: Subscriber's username (char [96], optional if MAC is present)

Sample command XML:

```
<USG COMMAND="USER_QUERY">
    <USER ID_TYPE="USER_NAME">jsmith</USER>
</USG>
```

or

```
<USG COMMAND="USER_QUERY">
    <USER ID_TYPE="MAC_ADDR">001122334455</USER>
</USG>
```

Response for the User Query Command

This is the response sent to User Query command. The response is an XML message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for User Query command
-->

<!ELEMENT MAC_ADDR (#PCDATA)>
<!ELEMENT USER_NAME (#PCDATA)>
<!ELEMENT PASSWORD (#PCDATA)>
<!ELEMENT EXPIRY_TIME (#PCDATA)>
<!ELEMENT ROOM_NUMBER (#PCDATA)>
<!ELEMENT PAYMENT_METHOD (#PCDATA)>
<!ELEMENT BILLING_STATUS (#PCDATA)>
<!ELEMENT DATA_VOLUME (#PCDATA)>
<!ELEMENT USG (MAC_ADDR, USER_NAME, PASSWORD, EXPIRY_TIME, ROOM_NUMBER,
PAYMENT_METHOD, BILLING_STATUS, DATA_VOLUME)>

<!ATTLIST USG
RESULT CDATA #REQUIRED
ID CDATA #REQUIRED
IP CDATA #REQUIRED
>
<!ATTLIST EXPIRY_TIME UNITS (SECONDS | MINUTES | HOURS | DAYS) #REQUIRED>
```

Where:
MAC_ADDR: Subscriber's MAC address (char [12])
USER_NAME: Subscriber's username (char [96])
PASSWORD: Subscriber's password (char [128])
EXPIRY_TIME: Expiry time
UNITS attribute: Either SECONDS, MINUTES, HOURS or DAYS
ROOM_NUMBER: Room number (Port-Location "Location" number) of access (char [8])
PAYMENT_METHOD: Either "PMS", "CREDIT_CARD", or blank if subscriber added by XML or by administrator
BILLING_STATUS: "DONE_OK" when 2-way PMS query is done and "DONE_ERROR" when the 2-way PMS query is not done.
DATA_VOLUME: data transferred by subscriber in Kbytes
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])

Sample Response XML:

```
<USG RESULT="OK" ID="ABC1234" IP="192.168.100.102">
    <MAC_ADDR>001122334455</MAC_ADDR>
    <USER_NAME>jsmith</USER_NAME>
    <PASSWORD>JSMITH6</PASSWORD>
    <EXPIRY_TIME UNITS=”SECONDS”>3600</EXPIRY_TIME>
    <ROOM_NUMBER>1234</ROOM_NUMBER>
    <PAYMENT_METHOD>PMS</PAYMENT_METHOD>
    <BILLING_STATUS>DONE_OK</BILLING_STATUS>
    <DATA_VOLUME>123456</DATA_VOLUME>
</USG>
```

3.13 Subscriber Query Current command

A query is made for information about a current subscriber. Information from that subscriber's entry in the "current table" is returned. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Current Subscriber Query command sent to NSE platforms
-->

<!ELEMENT MAC_ADDR (#PCDATA)>

<!ELEMENT USG (MAC_ADDR)>

<!ATTLIST USG COMMAND CDATA #REQUIRED>
```

Where:
COMMAND attribute: SUBSCRIBER_QUERY_CURRENT
MAC_ADDR: Subscriber's MAC address (char [12], required)

Sample command XML:

```
<USG COMMAND="SUBSCRIBER_QUERY_CURRENT">
    <MAC_ADDR>00508B74C8A6</MAC_ADDR>
</USG>
```

Response for the SUBSCRIBER_QUERY_CURRENT Command

This is the response sent to SUBSCRIBER_QUERY_CURRENT command. The response is an XML message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for Current Subscriber Query command
-->

<!ELEMENT MAC_ADDR (#PCDATA)>
<!ELEMENT IP_ADDR (#PCDATA)>
<!ELEMENT PORT_VLAN (#PCDATA)>
<!ELEMENT PORT_MODEM_MAC (#PCDATA)>
<!ELEMENT PORT_EMB_FDB (#PCDATA)>
<!ELEMENT ROOM_NUMBER (#CDATA)>
<!ELEMENT USER_NAME (#CDATA)>
<!ELEMENT USER_DEF1 (#CDATA)>
<!ELEMENT USER_DEF2 (#CDATA)>
<!ELEMENT MAX_BW_UP (#PCDATA)>
<!ELEMENT MAX_BW_DOWN (#PCDATA)>
<!ELEMENT THRUP_UP_IN (#PCDATA)>
<!ELEMENT THRUP_UP_OUT (#PCDATA)>
<!ELEMENT THRUP_DOWN_IN (#PCDATA)>
<!ELEMENT THRUP_DOWN_OUT (#PCDATA)>
<!ELEMENT AAA_STATE (#PCDATA)>
<!ELEMENT EXPIRY_TIME_SECS (#PCDATA)>
<!ELEMENT IDLE_TO_SECS (#PCDATA)>
<!ELEMENT BYTES_TX (#PCDATA)>
<!ELEMENT BYTES_RX (#PCDATA)>
<!ELEMENT PACKETS_TX (#PCDATA)>
<!ELEMENT PACKETS_RX (#PCDATA)>
<!ELEMENT PROXY_STATE (#PCDATA)>
<!ELEMENT AUTH_METHOD (#PCDATA)>
<!ELEMENT SMTP_REDIRECTION (#PCDATA)>
<!ELEMENT GROUP (#PCDATA)>
<!ELEMENT QOS_POLICY (#PCDATA)>
<!ELEMENT NAT_IP_ADDR (#PCDATA)>
<!ELEMENT IP_TYPE (#PCDATA)>
<!ELEMENT SUBSCRIBER_CURRENT (MAC_ADDR, IP_ADDR, PORT_VLAN?,
PORT_MODEM_MAC?,PORT_EMB_FDB?, ROOM_NUMBER?, USER_NAME, USER_DEF1?,
USER_DEF2?, MAX_BW_UP?, MAX_BW_DOWN?, THRUP_UP_IN?, THRUP_UP_OUT?,
THRUP_DOWN_IN?, THRUP_DOWN_OUT?, AAA_STATE, EXPIRY_TIME_SECS,
IDLE_TO_SECS, BYTES_TX, BYTES_RX, PACKETS_TX, PACKETS_RX, PROXY_STATE,
AUTH_METHOD, SMTP_REDIRECTION, GROUP?,QOS_POLICY?, NAT_IP_ADDR,
IP_TYPE?)>
<!ELEMENT USG (SUBSCRIBER_CURRENT?)>

<!ATTLIST USG
RESULT CDATA #REQUIRED
ID CDATA #REQUIRED
IP CDATA #REQUIRED
>
```

Where:

SUBSCRIBER_CURRENT (optional): Present if and only if result attribute of USG element has the value "OK".

MAC_ADDR: Subscriber's MAC address, exactly 12 hex-ascii characters in length.
IP_ADDR: Subscriber's IP address, up to 15 characters in length.
PORT_VLAN (optional): If subscriber is connected to a port on an 802.1Q concentrator, the port number to which he is connected.
PORT_MODEM_MAC (optional): If a subscriber is connected to a port on a Riverdelta 1000B or an Elastic Networks concentrator, this is the "modem MAC" address of the port to which he is connected.
PORT_EMB_FDB (optional): If a subscriber is connected to a port on an SNMP-based concentrator, this is the embellished port number to which he is connected.
NOTE: No more than one of the three elements PORT_VLAN, PORT_MODEM_MAC, and PORT_EMB_FDB will be present. The type that is present depends on which type of concentrator is being used on the NSE's subscriber-side network. If concentrators are not being used, or if a subscriber is directly connected, none of these elements will be present.
ROOM_NUMBER (optional): Room "number" or name (e.g., "Lobby") of access. Empty if no defined room is associated with the subscriber's port of access.
USER_NAME: Subscriber's username
USER_DEF1: User definable string (char [128])
USER_DEF2: User definable string (char [128])
MAX_BW_UP (optional): Effective maximum upstream bandwidth, in Kbps, for this subscriber. Empty if there is no effective limit for the subscriber. There is a limit only if the bandwidth management feature is enabled on the NSE AND there is a limit specified for the subscriber, either via an authFile entry or via RADIUS VSA.
MAX_BW_DOWN (optional): Effective maximum downstream bandwidth, in Kbps, for this subscriber. Empty if there is no effective limit for the subscriber. There is a limit only if the bandwidth management feature is enabled on the NSE AND there is a limit specified for the subscriber, either via an authFile entry or via RADIUS VSA.
THRUP_UP_IN (optional): The upstream data rate currently entering the NSE from this subscriber, in Kbps. Empty if the information is not presently available (e.g., throughput is not measured when bandwidth management is disabled).
THRUP_UP_OUT (optional): The upstream data rate currently exiting the NSE (on the network side) from this subscriber, in Kbps. Empty if the information is not presently available (e.g., throughput is not measured when bandwidth management is disabled).
THRUP_DOWN_IN (optional): The downstream data rate currently entering the NSE for this subscriber, in Kbps. Empty if the information is not presently available (e.g., throughput is not measured when bandwidth management is disabled).
THRUP_DOWN_OUT (optional): The downstream data rate currently exiting the NSE (on the subscriber side) for this subscriber, in Kbps. Empty if the information is not presently available (e.g., throughput is not measured when bandwidth management is disabled).
AAA_STATE: PENDING, VALID, UNKOWN, NO_ACCESS, TIMED_OUT, or AAA_OFF. TIMED_OUT will be returned if there is a session timer or idle timer which has expired. In this

case, the subscriber no longer has access to the network, and removal of the record from the current table is imminent.
EXPIRY_TIME_SECS: The amount of time left, in seconds, before the subscriber session times out. If there is no session timer for this session, this element will be empty. If the timer has already expired, the number shown here will be negative, and will reflect the number of seconds since the expiration. In this case, removal of the record from the current table is imminent.
IDLE_TO_SECS: The amount of idle time left, in seconds, before the subscriber's session is deemed to have ended due to inactivity. If there is no idle timer in effect for this session, the element will be empty. If the timer has already expired, the number shown here will be negative, and will reflect the number of seconds since the expiration. In this case, removal of the record from the current table is imminent.
BYTES_TX: Data transmitted by subscriber in bytes (64-bit value)
BYTES_RX: Data received by (delivered to) subscriber in bytes (64-bit value)
PACKETS_TX: Number of packets transmitted by subscriber.
PACKETS_RX: Number of packets received by (delivered to) subscriber.
PROXY_STATE: ON, OFF, or UNKNOWN.
AUTH_METHOD: Indicates the means by which a subscriber became authorized for network access. Values are NOT_AUTHORIZED (e.g., subscriber is still in Pending state), NOT_NEEDED (e.g., if AAA is turned off), RADIUS, CREDIT_CARD, PMS, TUNNELING, FREE_ROOM, ADMIN (if added to authorization database via the WMI, CLI, or SNMP), and XML. This element will be empty if the NSE software is unable to determine the authorization state of a subscriber.
NOTE: XML will be returned for any subscriber who was added to the database via an XML command, regardless of payment method element in that command. RADIUS, PMS, and CREDIT_CARD are returned only if the NSE itself has conducted the interaction with the corresponding server.
SMTP_REDIRECTION: Indicates whether or not SMTP redirection is effectively enabled for this subscriber, either ENABLED or DISABLED. "Effectively enabled" means that the subscriber's SMTP traffic will be redirected, i.e., the SMTP redirection feature is enabled globally on the NSE AND it is enabled for the individual subscriber (NOTE: There are separate global configuration parameters for enabling SMTP for well-configured and mis-configured subscribers).
GROUP (optional): Indicates whether subscriber is logged on to a group account, either TRUE or FALSE.
QOS_POLICY (optional): The name of the QOS policy in effect for this subscriber. Empty if no QOS policy is in effect.
NAT_IP_ADDR: The NAT IP address that has been assigned to this subscriber for DAT sessions. It will be reported as 0.0.0.0 if none has yet been assigned.
IP_TYPE (optional): If the subscriber has been assigned an address by the NSE, this parameter indicates whether the address was assigned from a public ("PUBLIC") or private ("PRIVATE") DHCP POOL. Note: This parameter indicates the IP_TYPE at the time of address assignment. Subsequent changes to DHCP Pools have no effect on this parameter.

RESULT attribute: OK or ERROR. See the DTD section entitled "Standard OK/ERROR Response" for the elements and attributes of an ERROR response. If no subscriber with the

specified MAC address is found, the error code 202 will be returned. If a matching entry is found in the current table but pertains to a "device" (as configured in the authFile), the same error code 202 will be returned, as such an entry does not pertain to a subscriber.
ID attribute: ID of the USG or NSE, exactly 6 hex-ascii characters in length.
IP attribute: IP address of the USG or NSE, up to 15 characters in length.

Note about optional elements: Elements specified above as optional will not be present if they pertain to an NSE feature that is not licensed on a particular NSE, or if they pertain to a feature that is not supported on the hardware platform on which the NSE is running. However, if a feature is licensed but is configured as disabled, the pertinent elements will be present in the response, but will be empty (contain no data). For example, if an NSE is not licensed for bandwidth management, the MAX_BW_UP and MAX_BW_DOWN elements will not be present in the response, but if bandwidth management is licensed but configured as disabled, these elements will be present but empty.

Implementation Notes for Portal/EWS Developers:

1) **Must gracefully handle/ignore elements not recognized**: In the future, as new NSE features are implemented or as new requirements arise for the *subscriber_query_current* command, new elements may be added to the response. An implementation must be prepared to gracefully ignore any unrecognized elements it may receive.
2) **Must gracefully handle missing optional elements**: Elements specified as optional in the DTD may or may not be present. An implementation must handle either case gracefully. See "note about optional elements" above for more detail.
3) **Must gracefully handle empty elements**: Many of the elements may be present but be empty of data, depending on NSE configuration and subscriber state. An implementation must be prepared to handle empty elements gracefully. See detailed element descriptions above and the "note about optional elements" above for explanation of the situations giving rise to empty elements.

Sample Responses (delivered without line feeds nor tabs/spaces):

The following example contains all of the elements, including one empty element :

```
<USG RESULT="OK" ID="ABC123" IP="192.168.100.102">
   <SUBSCRIBER_CURRENT>
      <MAC_ADDR>001122334455</MAC_ADDR>
      <IP_ADDR>10.0.0.12</IP_ADDR>
      <PORT_VLAN>101</PORT_VLAN>
      <ROOM_NUMBER>Lobby</ROOM_NUMBER>
      <USER_NAME><![CDATA[GeorgeIII]]></USER_NAME>
      <USER_DEF1><![CDATA[meeting room1]]></USER_DEF1>
      <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
      <MAX_BW_UP>1024</MAX_BW_UP>
      <MAX_BW_DOWN>1024</MAX_BW_DOWN>
      <THRUP_UP_IN>185</THRUP_UP_IN>
      <THRUP_UP_OUT>185</THRUP_UP_OUT>
      <THRUP_DOWN_IN>89</THRUP_DOWN_IN>
      <THRUP_DOWN_OUT>89</THRUP_DOWN_OUT>
      <AAA_STATE>VALID</AAA_STATE>
      <EXPIRY_TIME_SECS>40809</EXPIRY_TIME_SECS>
      <IDLE_TO_SECS></IDLE_TO_SECS>
      <BYTES_TX>45117330</BYTES_TX>
      <BYTES_RX>46169841</BYTES_RX>
      <PACKETS_TX>207328</PACKETS_TX>
      <PACKETS_RX>219564</PACKETS_RX>
      <PROXY_STATE>OFF</PROXY_STATE>
      <AUTH_METHOD>RADIUS</AUTH_METHOD>
      <SMTP_REDIRECTION>ENABLED</SMTP_REDIRECTION>
      <GROUP>FALSE</GROUP>
      <QOS_POLICY>RH_102</QOS_POLICY>
      <NAT_IP_ADDR>67.130.148.131</NAT_IP_ADDR>
      <IP_TYPE>PRIVATE</IP_TYPE>

   </SUBSCRIBER_CURRENT>
</USG>
```

The following example contains the minimal set of elements, illustrating the case when all relevant NSE features are unlicensed :

```
<USG RESULT="OK" ID="ABC123" IP="192.168.100.102">
    <SUBSCRIBER_CURRENT>
        <MAC_ADDR>001122334455</MAC_ADDR>
        <IP_ADDR>10.1.1.56</IP_ADDR>
        <USER_NAME><![CDATA[MBM]]></USER_NAME>
        <USER_DEF1><![CDATA[meeting room1]]></USER_DEF1>
        <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
        <AAA_STATE>VALID</AAA_STATE>
        <EXPIRY_TIME_SECS>17567</EXPIRY_TIME_SECS>
        <IDLE_TO_SECS>297</IDLE_TO_SECS>
        <BYTES_TX>852677</BYTES_TX>
        <BYTES_RX>1983451</BYTES_RX>
        <PACKETS_TX>10342</BYTES_TX>
        <PACKETS_RX>33986</BYTES_RX>
        <PROXY_STATE>ON</PROXY_STATE>
        <AUTH_METHOD>XML</AUTH_METHOD>
        <SMTP_REDIRECTION>ENABLED</SMTP_REDIRECTION>
        <NAT_IP_ADDR>67.130.148.131</NAT_IP_ADDR>
    </SUBSCRIBER_CURRENT>
</USG>
```

3.14 Subscriber Query Auth Command

A query is made for information about a subscriber configuration saved in the authorized database. Information from that subscriber's entry in the "auth table" is returned. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Subscriber Query Auth command sent to NSE platforms
-->

<!ELEMENT MAC_ADDR  (#PCDATA)>
<!ELEMENT USER_NAME (#PCDATA)>

<!ELEMENT USG (MAC_ADDR | USER_NAME)>

<!ATTLIST USG COMMAND CDATA #REQUIRED>
```

Where:

COMMAND attribute: SUBSCRIBER_QUERY_AUTH
MAC_ADDR: Subscriber's MAC address (char [12], required)
or
USER_NAME: Subscriber's name. (char [96], required)

Sample command XML:

```
<USG COMMAND="SUBSCRIBER_QUERY_AUTH">
    <MAC_ADDR> 0010A4BABD5C</MAC_ADDR>
</USG>
```

Response for the SUBSCRIBER_QUERY_AUTH Command

This is the response sent to *SUBSCRIBER_QUERY_AUTH* command. The response is an XML message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for Subscriber Query Auth command
-->

<!ELEMENT MAC_ADDR   (#PCDATA)>
<!ELEMENT USER_NAME (#CDATA)>
<!ELEMENT IP_ADDR      (#PCDATA)>
<!ELEMENT SUBNET      (#PCDATA)>
<!ELEMENT EXPIRY_TIME_SECS (#PCDATA)>
<!ELEMENT EXPIRED (#PCDATA)>
<!ELEMENT AMT_PAID     (#PCDATA)>
<!ELEMENT AMT_LEFT     (#PCDATA)>
<!ELEMENT USER_DEF1 (#CDATA)>
<!ELEMENT USER_DEF2 (#CDATA)>
<!ELEMENT AUTH_METHOD (#PCDATA)>
<!ELEMENT COUNT_DOWN (#PCDATA)>
<!ELEMENT COUNTING_DOWN (#PCDATA)>
<!ELEMENT IP_TYPE (#PCDATA)>
<!ELEMENT MAX_BW_UP  (#PCDATA)>
<!ELEMENT MAX_BW_DOWN (#PCDATA)>
<!ELEMENT BILLING_PLAN (#PCDATA)>
<!ELEMENT QOS_POLICY   (#PCDATA)>
<!ELEMENT SMTP_REDIRECTION   (#PCDATA)>

<!ELEMENT SUBSCRIBER_AUTH (MAC_ADDR, USER_NAME, IP_ADDR, SUBNET,
EXPIRY_TIME_SECS, EXPIRED, AMT_PAID, AMT_LEFT, USER_DEF1, USER_DEF2,
AUTH_METHOD, COUNT_DOWN, COUNTING_DOWN, IP_TYPE?, MAX_BW_UP?,
MAX_BW_DOWN?, BILLING_PLAN,  QOS_POLICY?, SMTP_REDIRECTION)>
<!ELEMENT USG (SUBSCRIBER_AUTH?)>

<!ATTLIST USG
RESULT CDATA #REQUIRED
ID CDATA #REQUIRED
IP CDATA #REQUIRED
>
```

Where:

MAC_ADDR: Subscriber's MAC address, exactly 12 hex-ascii characters in length. An empty string is returned if subscriber was added by name.
USER_NAME: Subscriber's username, up to 96 hex-ascii characters. An empty string is returned if subscriber was added by MAC.
IP_ADDR: Subscriber's IP address, up to 15 characters in length. (May not reflect the correct IP address assigned to this subscriber.) This value may change at the IP update time.
SUBNET: Subscriber's subnet.
EXPIRY_TIME_SECS: The amount of time left, in seconds, before the subscriber account times out. An empty string will be returned if there is no expiration time or this subscriber expired.

EXPIRED: String indicating if this subscriber has expired. (“TRUE” or “FALSE”).
AMT_PAID: Amount paid by the user of this account.
AMT_LEFT: Amount left on this account. For x-over-y subscribers this value does not reflect the
USER_DEF1: User definable string (char [128])
USER_DEF2: User definable string (char [128])
The actual amount left on this account, which will be updated at the logout time.
AUTH_METHOD: String indicating by what method the subscriber was added to the authorized persistent database. Values are: “PMS”, “CREDIT_CARD”, “XML”, “ADMIN”.
Radius and post-paid PMS subscribers will not appear in the authorized database.
Some other methods of authorization may be added in the future and the users of this command should be prepared to handle such cases.
Note: “XML” will be returned for subscribers that were added via XML commands, regardless of the payment method.

COUNT_DOWN: String indicating if Count-down starts after Login for this subscriber; (“ENABLED” or “DISABLED”).
COUNTING_DOWN: String indicating if the time is running down for this subscriber. (“TRUE” or “FALSE”).
IP_TYPE (optional): String indicating what kind of IP the user is authorized to use. (“PRIVATE” or “PUBLIC”).
MAX_BW_UP (optional): Configured maximum upstream bandwidth, in Kbps, for this subscriber. An empty string will be returned if this parameter was not configured when the account was created – meaning UNLIMITED.
MAX_BW_DOWN (optional): Configured maximum downstream bandwidth, in Kbps, for this subscriber. An empty string will be returned if this parameter was not configured when the account was created – meaning UNLIMITED.
BILLING_PLAN: Plan number associated with this account. An empty string is returned if there is no associated plan for this subscriber.
QOS_POLICY (optional): QoS policy associated with this account, up to 16 characters in length. An empty string will be returned if no policy is assigned to this subscriber.
SMTP_REDIRECTION: String indicating if the SMTP protocol redirection is enabled for this subscriber. (“ENABLED”, or “DISABLED”) Note: This does not take into account a global status of SMTP redirect. Rather, how the individual subscriber was configured.

RESULT attribute: “OK” or “ERROR”. See the DTD section entitled “Standard OK/ERROR Response” for the elements and attributes of an ERROR response.

If specified MAC address found is not a subscriber an error 202 will be returned “Unknown user MAC address”, along with the syslog message “User: ‘*MAC*’ is a Device”.
When subscriber query by username finds record for a group account, an error 201 will be returned “Unknown user name”, along with the syslog msg “User: ‘*Name*’ is a Group Account”.

ID attribute: ID of NSE, exactly 6 hex-ascii characters in length.
IP attribute: IP address of the NSE, up to 15 characters in length.

Note about optional elements:

Elements specified as optional will not be present if they are not licensed on a particular NSE or if they are not implemented on a hardware platform on which the NSE is running. However, if the feature is licensed but was not configured for the particular subscriber, the element will be present in the response but will contain no data. For example, if the quality of service on a particular unit is licensed but user did not select policy during configuration, the element QOS_POLICY will be present but will contain en empty string.

Implementation Notes for Portal/EWS Developers:

1) **Must gracefully ignore elements not recognized:** In the future, as new NSE features are implemented or as new requirements arise for the *subscriber_query_auth* command, new elements may be added to the response. An implementation must be prepared to gracefully ignore any unrecognized elements it may receive.
2) **Must gracefully handle missing optional elements:** Elements specified as optional in the DTD may or may not be present. An implementation must handle either case gracefully. See "note about optional elements" above for more detail.
3) **Must gracefully handle empty elements:** Many of the elements may be present but be empty of data, depending on NSE configuration and subscriber state. An implementation must be prepared to handle empty elements gracefully. See detailed element descriptions above and the "note about optional elements" above for more details.

Sample Response XML:

```
<USG RESULT="OK" ID="ABC123" IP="192.168.100.102">
    <SUBSCRIBER_AUTH>
        <MAC_ADDR>001122334455</MAC_ADDR>
        <USER_NAME><![CDATA[Gonzales]]></USER_NAME>
        <IP_ADDR>10.0.0.12</IP_ADDR>
        <SUBNET></SUBNET>
        <EXPIRY_TIME_SECS>40809</EXPIRY_TIME_SECS>
        <AMT_PAID>678.55</AMT_PAID>
        <AMT_LEFT>16.35</AMT_LEFT>
        <USER_DEF1><![CDATA[meeting room1]]></USER_DEF1>
        <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
        <AUTH_METHOD>RADIUS</AUTH_METHOD>
        <COUNT_DOWN>ENABLED</COUNT_DOWN>
        <COUNTING_DOWN>TRUE</COUNTING_DOWN>
        <IP_TYPE>PRIVATE</IP_TYPE>
        <MAX_BW_UP>512</MAX_BW_UP>
        <MAX_BW_DOWN>1024</MAX_BW_DOWN>
        <BILLING_PLAN>5</BILLING_PLAN>
        <QOS_POLICY>RH_102</QOS_POLICY>
        <SMTP_REDIRECTION>ENABLED</SMTP_REDIRECTION>
    </SUBSCRIBER_AUTH>
</USG>
```

3.15 User Authorize Command

A subscriber's identity, specified by his MAC address, is checked against the authorization table. If the subscriber is found in the MAC authorization table, **VALID_USER** is returned along with the subscriber's authorization method: **PMS** or **CREDIT_CARD**. If the subscriber is not found, **INVALID_USER** will be returned. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Authorize command sent to NSE
-->

<!ELEMENT USG (EMPTY)>

<!ATTLIST USG
COMMAND CDATA #REQUIRED
        MAC_ADDR CDATA #REQUIRED
>
```

Where:
COMMAND : "USER_AUTHORIZE"
MAC_ADDR attribute: Subscriber's MAC address (char [12])

Sample command XML:

```
<USG COMMAND="USER_AUTHORIZE" MAC_ADDR="1A2B3C4D5E6F">
</USG>
```

Response for the User Authorize Command

This is the response sent for User Authorize command. The response is an XML message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for User Authorize command
-->

<!ELEMENT STATUS (#PCDATA)>
<!ELEMENT PAYMENT_METHOD (#PCDATA)>
<!ELEMENT USG (STATUS, PAYMENT_METHOD)>

<!ATTLIST USG
RESULT CDATA #REQUIRED
ID CDATA #REQUIRED
IP CDATA #REQUIRED
>
```

Where:
STATUS: “VALID_USER” or “INVALID USER”
PAYMENT_METHOD: “PMS” or “CREDIT_CARD”
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])

Sample Response XML:

```
<USG RESULT="OK" ID="ABC123" IP="192.168.100.102">
    <STATUS>VALID_USER</STATUS>
    <PAYMENT_METHOD>PMS</PAYMENT_METHOD>
</USG>
```

3.16 User Purchase Command

A subscriber's e-commerce or special service purchase is to be charged. Currently, the only option is to charge the subscriber's bill via the PMS system. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Purchase command sent to NSE
-->
<!ELEMENT ITEM_CODE (#PCDATA)>
<!ELEMENT ITEM_DESCRIPTION (#PCDATA)>
<!ELEMENT ITEM_AMOUNT (#PCDATA)>
<!ELEMENT ITEM_TAX (#PCDATA)>
<!ELEMENT ITEM_TOTAL (#PCDATA)>
<!ELEMENT REAL_NAME(#PCDATA)>
<!ELEMENT MAC_ADDRESS(#PCDATA)>
<!ELEMENT REG_NUMBER(#PCDATA)>
<!ELEMENT TRANS_ID(#PCDATA)>
<!ELEMENT CC_SUFFIX(#PCDATA)>
<!ELEMENT CC_EXPIRATION(#PCDATA)>
<!ELEMENT WFB_BUNDLED(#PCDATA)>
<!ELEMENT REVENUE_CENTER(#PCDATA)>

<!ELEMENT USG (ITEM_CODE, ITEM_DESCRIPTION, ITEM_AMOUNT, ITEM_TAX,
ITEM_TOTAL, REAL_NAME?, MAC_ADDRESS?, REG_NUMBER?, TRANS_ID?, CC_SUFFIX?,
CC_EXPIRATION?, WFB_BUNDLED?, REVENUE_CENTER?)>

<!ATTLIST USG
 COMMAND CDATA #REQUIRED
          ROOM_NUMBER CDATA #REQUIRED
>

<!ATTLIST WFB_BUNDLED WFB_OPTION (A | B | C | D) #IMPLIED>
```

Where:

COMMAND attribute: USER_PURCHASE
ROOM_NUMBER attribute: Room number (Port-Location "Location" number), (char [8])
ITEM_CODE: Code of the item being purchased
ITEM_DESCRIPTION: Description of the item
ITEM_AMOUNT: Item amount
ITEM_TAX; Item tax
ITEM_TOTAL: Item total
REAL_NAME: Name in the PMS DATABASE Only needed for 2-way PMS
MAC_ADDRESS: MAC Address of the Subscriber Only needed for Post Paid PMS
REG_NUMBER: Registration number required for 2-way FIAS PMS
CC_SUFFIX: (optional): Last 4 Digits of the Credit Card for Marriott WFB PMS Verification.

CC_EXPIRATION: (optional): Expiration Date on the Credit Card for Marriott WFB PMS Verification. Format = MMYY.
WFB_BUNDLED: (optional): WFB Bundle Bill. 0 = Charge 1 = Bundle
WFB_OPTION attribute: Either A, B, C or D
TRANS_ID: (optional): (unsigned Integer) Used to match commands with USER_STATUS messages. Information entered here will be mirrored on the USER_STATUS messages.
REVENUE_CENTER: (optional): 3 Digits to specify the Revenue Center for MICROS PMS, or 2 Digits to specify Revenue Code for Marriott WFB and Marriott FOSSE.

Sample command XML:

```
<USG COMMAND="USER_PURCHASE" ROOM_NUMBER="1234">
    <ITEM_CODE>123</ITEM_CODE>
    <ITEM_DESCRIPTION>Tooth Brush</ITEM_DESCRIPTION>
    <ITEM_AMOUNT>2.49</ITEM_AMOUNT>
    <ITEM_TAX>0.21</ITEM_TAX>
    <ITEM_TOTAL>2.70</ITEM_TOTAL>
    <REAL_NAME>Smith</REAL_NAME>
    <MAC_ADDRESS>010203040506</MAC_ADDRESS>
    <REG_NUMBER>12345</REG_NUMBER>
    <TRANS_ID>12345</TRANS_ID>
    <CC_SUFFIX>1234</CC_SUFFIX>
    <CC_EXPIRATION>1209</CC_EXPIRATION>
    <WFB_BUNDLED WFB_OPTION=”A”>1</WFB_BUNDLED>
    <REVENUE CENTER>1</REVENUE_CENTER>
</USG>
```

Response for the User Purchase Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see “Standard OK/ERROR Response” section for DTD definition).

3.17 PMS Pending Transaction Command

NOTE: This command should be sent as a POST to the following address:

http(s)://NSE_URI/api/pmsRedirector/v1/pendingTransaction

Submit a pending PMS transaction to be processed by the PMS Serial Redirector. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines PMS Pending Transaction command
-->

<!ELEMENT DATA (#PCDATA)>
<!ELEMENT TRANSACTION_ID (#PCDATA)>
<!ELEMENT P_TRANSACTION (DATA, TRANSACTION_ID)>
<!ELEMENT USG (P_TRANSACTION)>

<!ATTLIST USG COMMAND CDATA #REQUIRED>
```

Where:

COMMAND attribute: PMS_PENDING_TRANSACTION
TRANSACTION_ID: (optional): (32 bit unsigned Integer) Used to match commands with PMS_TRANSACTION_RESPONSE messages. Information entered here will be mirrored on the PMS_TRANSACTION_RESPONSE messages.
DATA: The data that will be sent to the attached PMS system. Before sending, the data is framed with an ETX (hex 02) and an STX (hex 03) and appended with a checksum.

Sample command XML:

```
<USG COMMAND="PMS_PENDING_TRANSACTION">
    <P_TRANSACTION>
        <TRANSACTION_ID>123445</TRANSACTION_ID>
        <DATA>PR|PI1008  |DA110629|TI131100|P#0001|CTPlan A|</DATA>
    </P_TRANSACTION>
</USG>
```

Response for the PMS Pending Transaction Command

The response to this command will indicate whether or not the command was successfully queued on the NSE for processing.

An HTTP response code of 200 indicates success. The DTD of a successful response XML is:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines successful response for PMS Pending Transaction command
-->

<!ELEMENT ID (#PCDATA)>
<!ELEMENT LINK_STATE (#PCDATA)>
<!ELEMENT DATA (#PCDATA)>
<!ELEMENT TRANSACTION_ID (#PCDATA)>
<!ELEMENT P_TRANSACTION (ID, LINK_STATE, DATA, TRANSACTION_ID)>
<!ELEMENT USG (P_TRANSACTION)>

<!ATTLIST USG COMMAND CDATA #REQUIRED>
```

Where:

COMMAND attribute: PMS_PENDING_TRANSACTION
ID: (32 bit unsigned Integer) A unique ID that can be used to identify the transaction in an “xxx” or an “xxxx” command.
LINK_STATE: (optional): contains the value “DOWN” as is present only if the link to the attached PMS system is down.
TRANSACTION_ID: (optional): (32 bit unsigned Integer) The TRANSACTION_ID from the corresponding command.
DATA: The DATA from the corresponding command.

Sample successful response XML:

```
<USG COMMAND="PMS_PENDING_TRANSACTION" VERSION="1.0">
  <P_TRANSACTION URI="/api/pmsRedirector/v1/pendingTransaction/2">
    <ID>2</ID>
    <LINK_STATE>DOWN</LINK_STATE>
    <TRANSACTION_ID>123445</TRANSACTION_ID>
    <DATA><![CDATA[PR|PI1008  |DA110629|TI131100|P#0001|CTPlan A|]]></DATA>
  </P_TRANSACTION>
</USG>
```

An HTTP response code of 500 is used to indicate that the request failed. An XML error response will be send to the web server. The DTD of the error response XML is:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines successful response for PMS Pending Transaction command
-->

<!ELEMENT ERROR_CODE (#PCDATA)>
<!ELEMENT ERROR_DESCRIPTION (#PCDATA
<!ELEMENT USG (ERROR_CODE, ERROR_DESCRIPTION)>

<!ATTLIST USG RESULT CDATA #REQUIRED>
```

Where:

Result attribute: “ERROR”
ERROR_CODE: Indicates the numberic error code. 1 indicates an XML syntax error in the command, 3 indicates that the PMS transaction queue on the NSE is full. All other error conditions are indicated by a value of 4

ERROR_DESCRITPION: A description of the error corresponding to the ERROR_CODE. The description for the error code of 1 is “Syntax error”. The description for the error code 3 is “Collection full”. The description for the error code 4 is “Unknown error”

Sample error response XML:

```
USG RESULT=”ERROR” VERSION=”1.0”>
        <ERROR_CODE>3</ERROR_CODE>
        <ERROR_DESCRIPTION>Collection Full</ERROR_DESCRIPTION>
 </USG>
```

4. Room Administration Commands

NOTE: The commands listed in this section should be sent as a POST to one of the following addresses:

http://NSE_URI:1111/usg/command.xml
https://NSE_URI:1112/usg/command.xml

Please note the port difference between standard and secure transmissions.

4.1 Room Set Access Command

This command will be sent by the Administrator to the NSE when room access needs to be set. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Room Set Access command sent to NSE
-->

<!ELEMENT ACCESS_MODE (#PCDATA)>

<!ELEMENT USG( ACCESS_MODE )>

<!ATTLIST USG COMMAND CDATA #REQUIRED
   ROOM_NUMBER  CDATA #REQUIRED
>
```

Where:
COMMAND attribute: “ROOM_SET_ACCESS”
ROOM_NUMBER attribute: Room number (Port-Location "Location" number), (char [8])
ACCESS_MODE: Type of access ROOM_OPEN, ROOM_CHARGE, or ROOM_BLOCK

Sample command XML:

```
<USG COMMAND=”ROOM_SET_ACCESS” ROOM_NUMBER=”1234”>
    <ACCESS_MODE>ROOM_OPEN</ACCESS_MODE>
</USG>
```

Response for the Set Room Access Command

Standard: As a response to this command, the web server will get an acknowledgement XML message from the NSE (OK or ERROR, see “Standard OK/ERROR Response” section for DTD definition).

4.2 Room Query Access Command

This command will be sent by the Administrator to the NSE when there is a need to query the access status of a room. This is the XML command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  DTD defines Room Query Access command sent to NSE
-->
<!ELEMENT USG(EMPTY)>

<!ATTLIST USG COMMAND CDATA #REQUIRED
   ROOM_NUMBER  CDATA #REQUIRED
>
```

Where:
COMMAND attribute: "ROOM_QUERY_ACCESS"
ROOM_NUMBER attribute: Room number (Port-Location "Location" number), (char [8])

Sample command XML:

```
<USG COMMAND=”ROOM_QUERY_ACCESS” ROOM_NUMBER=”1234”>
</USG>
```

Response for the Room Query Access Command

This is the response sent for Room Query Access command. The response is an XML message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for Room Query Access command
-->

<!ELEMENT ERROR_NUM (#PCDATA)>
<!ELEMENT ERROR_DESC (#PCDATA)>
<!ELEMENT ACCESS_MODE (#PCDATA)>
<!ELEMENT ROOM_NUMBER(#PCDATA)>

<!ELEMENT USG (ERROR_NUM?, ERROR_DESC?,ACCESS_MODE?,ROOMNUMBER?)>

<!ATTLIST USG
    RESULT CDATA #REQUIRED
           ID CDATA #REQUIRED
           IP CDATA #REQUIRED
>
```

Where:
RESULT attribute: ‘OK’ or ‘ERROR’. In case of ‘ERROR’, ERORR_NUM and ERROR_DESC elements must be present.
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])
ERROR_NUM: ‘102’or ‘200 ‘, present only when RESULT is ‘ERROR’.
ERROR_DESC: ‘Required attribute is missing’ when ERROR_NUM is ‘102’, ‘Unknown room number’ when ERROR_NUM is ‘200’.

Sample OK XML:

```
<USG RESULT="OK" ID="ABC123" IP="192.168.100.102">
        <ROOM_NUMBER>1234</ROOM_NUMBER>
        <ACCESS_MODE>ROOM_OPEN</ACCESS_MODE>
</USG>
```

5. Standard Response

5.1 Standard OK/ERROR Response

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Standard Response from NSE
-->

<!ELEMENT ERROR_NUM (#PCDATA)>
<!ELEMENT ERROR_DESC (#PCDATA)>

<!ELEMENT USG (ERROR_NUM, ERROR_DESC)?>
<!ATTLIST USG
   RESULT CDATA #REQUIRED
          ID CDATA #REQUIRED
          IP CDATA #REQUIRED
>
```

Where:
RESULT attribute: 'OK' or 'ERROR'. In case of 'ERROR', ERORR_NUM and ERROR_DESC elements will be present.
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])
ERROR_NUM: present only when RESULT is 'ERROR' (see Response Errors for XML Command section).
ERROR_DESC: (see Response Errors for XML Command section).

Sample OK XML:

```
<USG RESULT="OK" ID="ABC123" IP="192.168.100.102">
</USG>
```

Sample ERROR XML:

```
<USG RESULT="ERROR" ID="ABC123" IP="192.168.100.102">
    <ERROR_NUM>102</ERROR_NUM>
    <ERROR_DESC>Required attribute is missing</ERROR_DESC>
</USG>
```

5.2 Response Errors for XML Command

Error No.	Error Description String
100	Parsing error
101	Unrecognized command
102	Required attribute is missing
103	Required data is missing
200	Unknown room number
201	Unknown user name
202	Unknown user MAC address
203	Wrong password
204	User name already used
205	Too many subscribers
206	Unable to provide all requested data
207	AAA internal error (when AAA is not configured correctly for the command request)
208	Wrong Plan Number
209	User is already valid
210	Specified valid-until time is invalid
211	Specified DHCP subnet does not exist
300	User RADIUS account not found
301	User RADIUS authorization denied
302	User PMS authorization denied
303	Unsupported payment method
304	MAC Address does not belong to room location

6. User Status Messages for Radius and 2-way PMS

6.1 User Status Message for Radius Login/Logout

The NSE sends this message to the Portal Page web server when the subscriber's status changes. This is the XML command message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Status Message sent from NSE
-->

<!ELEMENT SUB_MAC_ADDR (#PCDATA)>
<!ELEMENT SUB_STATUS (#PCDATA)>
<!ELEMENT SUB_USER_NAME (#PCDATA)>
<!ELEMENT PORTAL_SUB_ID (#PCDATA)>

<!ELEMENT USG (SUB_MAC_ADDR, SUB_USER_NAME, SUB_STATUS, PORTAL_SUB_ID)>
<!ATTLIST USG
   COMMAND CDATA #REQUIRED
           ID CDATA #REQUIRED
           IP CDATA #REQUIRED
>
```

Where:
COMMAND attribute: 'USER_STATUS'
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])
SUB_MAC ADDRESS: Subscriber's MAC address (char [12])
SUB_STATUS: One of: 'RADIUS_LOGIN', 'RADIUS_LOGIN_ACCEPT', 'RADIUS_LOGIN_REJECT', 'RADIUS_LOGIN_ERROR', 'RADIUS_LOGIN_TIMEOUT','RADIUS_LOGOUT', 'RADIUS_LOGOUT_PORTAL_RESET', 'RADIUS_LOGOUT_IDLE_TIMEOUT', 'RADIUS_LOGOUT_SESSION_TIMEOUT', 'RADIUS_LOGOUT_USER_REQUEST', or 'RADIUS_LOGOUT_ADMIN_RESET' (char [35])
'SUB_USER_NAME: Subscriber's Username (char [96])
PORTAL_SUB_ID: Some unique identifier that the Portal Web Server can send to the NSE, which will be sent back on responses for that request. (char [36])

Status Message	Description
RADIUS_LOGIN	Default Login Response if no match for other RADIUS_LOGIN messages, i.e. Access-Challenges will reproduce this message.
RADIUS_LOGIN_ACCEPT	Login by XML or IWS (Internal Web Server) Login or HTML GET (SSL or non-SSL)
RADIUS_LOGIN_REJECT	Login Reject
RADIUS_LOGIN_ERROR	An error occurred.
RADIUS_LOGIN_TIMEOUT	Login Timeout
RADIUS_LOGOUT	Default Logout Response if no match for other RADIUS_LOGOUT messages
RADIUS_LOGOUT_PORTAL_RESET	XML Logout
RADIUS_LOGOUT_IDLE_TIMEOUT	Idle Timeout
RADIUS_LOGOUT_SESSION_TIMEOUT	Session Timeout
RADIUS_LOGOUT_USER_REQUEST	ICC (Information Control Console) or http://1.1.1.1 Logout
RADIUS_LOGOUT_ADMIN_RESET	Logout by Administrator (deleted from NSE administration)

Sample command XML:

```
<USG COMMAND="USER_STATUS" ID="ABC123" IP="192.168.100.102">
    <SUB_MAC_ADDR>001122334455</SUB_MAC_ADDR>
    <SUB_USER_NAME>jsmith</SUB_USER_NAME>
    <SUB_STATUS>RADIUS_LOGIN_ACCEPT</SUB_STATUS>
    <PORTAL_SUB_ID>12345678-1234-1234-1234-123456789012</PORTAL_SUB_ID>
</USG>
```

6.2 PMS User Status

The NSE sends this message to the Portal Page web server when the subscriber's Purchases time with MICROS PMS. This is the XML command message with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines User Status Message sent from NSE
-->

<!ELEMENT SUB_MAC_ADDR (#PCDATA)>
<!ELEMENT SUB_USER_NAME (#PCDATA)>
<!ELEMENT TRANSACTION_ID (#PCDATA)>
<!ELEMENT SUB_STATUS (#PCDATA)>
<!ELEMENT FAIL_REASON (#PCDATA)>
<!ELEMENT PMS_ERROR (#PCDATA)>

<!ELEMENT USG (SUB_MAC_ADDR, SUB_USER_NAME, TRANSACTION_ID, SUB_STATUS,
FAIL_REASON, PMS_ERROR)>
<!ATTLIST USG
   COMMAND CDATA #REQUIRED
           ID CDATA #REQUIRED
           IP CDATA #REQUIRED
>
```

Where:
COMMAND attribute: 'USER_STATUS'
ID attribute: ID of the NSE (char [6])
IP attribute: IP address of the NSE (char [18])

SUB_MAC_ADDRESS: Subscriber's MAC address (char [12])
SUB_USER_NAME: Subscriber's user name
TRANSACTION_ID (optional): The TRANS_ID sent in the USER_PAYMENT or USER_PURCHASE
SUB_STATUS: One of: "PMS_INVALID_CREDENTIALS","PMS_COMPLETED", "PMS_FAILED", "PMS_POST_PAID"

Status Message	Description
PMS_INVALID_CREDENTIALS	The PMS Name or Room number doesn't match anything that the user entered.
PMS_COMPLETED	The PMS Transaction was successful and a POST should have been sent and accepted by the PMS System
PMS_FAILED	The transaction was denied, transaction failed for unspecified reasons, or the NSE got something from PMS we did not expect.
PMS_POST_PAID	The PMS on the NSE is set to POST Paid and the initial verification of the users credentials completed successfully but did not POST a bill.

FAIL_REASON (optional): MOP_CA (cash payment), MOP_SC (signed charges), or NO_POST (NPY shown in the PL response to the PR message)
PMS_ERROR (optional): Error Code returned by PMS

Sample command XML:

```
<USG COMMAND="USER_STATUS" ID="0164b3" IP="192.168.100.102">
    <SUB_MAC_ADDR>00:11:22:33:44:55</SUB_MAC_ADDR>
    <SUB_USER_NAME>gray</SUB_USER_NAME>
    <TRANSACTION_ID>1234</TRANSACTION_ID>
    <SUB_STATUS>PMS_POST_PAID</SUB_STATUS>
</USG>
<USG COMMAND="USER_STATUS" ID="0164b3" IP="192.168.100.102">
    <SUB_MAC_ADDR>00:11:22:33:44:55</SUB_MAC_ADDR>
    <SUB_USER_NAME>gray</SUB_USER_NAME>
    <SUB_STATUS>PMS_FAILED</SUB_STATUS>
    <FAIL_REASON>MOP_CA</FAIL_REASON>
</USG>
<USG COMMAND="USER_STATUS" ID="0164b3" IP="192.168.100.102">
    <SUB_MAC_ADDR>00:11:22:33:44:55</SUB_MAC_ADDR>
    <SUB_USER_NAME>gray</SUB_USER_NAME>
    <SUB_STATUS>PMS_FAILED</SUB_STATUS>
    <PMS_ERROR>11</PMS_ERROR>
</USG>
```

For reference, the following table provides a summary of the interaction between the PMS payment type (Pre-pay or Post-pay), the "Free for PMS User" option in a billing plan, and the NP (No Post) flag returned in the PL response to the PR message:

Parameter Combinations			Expected Results		
PMS Pay Type	**"Free for PMS" in billing plan**	**No Post (NP) value returned by PMS**	**PMS Activity (assumes valid user)**	**Internet access**	**XML Portal Post Message(s)**
Pre-pay	Disabled	N	query, post	Granted	PMS_COMPLETED
Pre-pay	Disabled	Y	reject	Rejected	PMS_FAILED
Pre-pay	Enabled	N	query only	Granted	PMS_COMPLETED
Pre-pay	Enabled	Y	query only	Granted	PMS_COMPLETED
Post-pay	Disabled	N	query, post later	Granted	PMS_POST_PAID at login, PMS_COMPLETED at logoff or expiration (when charge posted)
Post-pay	Disabled	Y	reject	Rejected	PMS_FAILED
Post-pay	Enabled	N	query only	Granted	PMS_POST_PAID
Post-pay	Enabled	Y	query only	Granted	PMS_POST_PAID

7. PMS Redirector Status/Response Messages

The messages defined in this section are used by the NSE's PMS Redirector feature to send PMS responses and status to the Portal Page web server.

Each message is numbered (the SEQUENCE_NUMBER element). The Portal Page web server can use this sequence number to detect lost messages indicating some sort of communication problem.

7.1 PMS Link Status

The PMS Link Status Message is used by the NSE to indicate, to the Portal Page web server, whether or not the link layer (as defined by the FIAS specification) between the NSE and the PMS system is correctly functioning (i.e. UP or DOWN).

The NSE sends this message to the Portal Page web server whenever the NSE detects a change in the PMS Link Status.

The DTD for this status message is the following:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines PMS Link Status Message sent from NSE
-->

<!ELEMENT NSE_ID (#PCDATA)>
<!ELEMENT SEQUENCE_NUMBER (#PCDATA)>
<!ELEMENT LINK_STATUS (#PCDATA)>

<!ELEMENT USG (NSE_ID, SEQUENCE_NUMBER, LINK_STATUS)>
<!ATTLIST USG
   COMMAND CDATA #REQUIRED
   VERSION CDATA #REQUIRED
>
```

Where:
COMMAND attribute: 'PMS_LINK_STATUS'
VERSION attribute: '1.0'
NSE_ID: ID of the NSE (char [6])
SEQUENCE_NUMBER: The sequence number of the message
LINK_STATUS: The status of the PMS link, "UP" or "DOWN"

Sample PMS Link Status Message:

```
<USG COMMAND="PMS_LINK_STATUS" VERSION="1.0" >
    <NSE_ID>112233</NSE_ID>
    <SEQUENCE_NUMBER>4871</SEQUENCE_NUMBER>
    <LINK_STATUS>UP</LINK_STATUS>
</USG>
```

7.2 PMS Transaction Response

For each PMS Pending transaction command posted to the NSE (refer to section 3.17), one or responses will be generated by the PMS system. Each of these responses will be encapsulated in a PMS Transaction Response message and sent to the Portal Page web server.

The DTD for this message is the following:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines PMS Transaction Response sent from NSE
-->

<!ELEMENT NSE_ID (#PCDATA)>
<!ELEMENT SEQUENCE_NUMBER (#PCDATA)>
<!ELEMENT TRANSACTION_ID (#PCDATA)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT ERROR_CODE (#PCDATA)>
<!ELEMENT COMPLETION_CODE (#PCDATA)>
<!ELEMENT DATA (#PCDATA)>

<!ELEMENT USG (NSE_ID, SEQUENCE_NUMBER, TRANSACTION_ID, ID, ERROR_CODE,
COMPLETION_CODE, DATA?)>
<!ATTLIST USG
   COMMAND CDATA #REQUIRED
   VERSION CDATA #REQUIRED
>
```

Where:
COMMAND attribute: 'PMS_TRANSACTION_RESPONSE'
VERSION attribute: '1.0'
NSE_ID: ID of the NSE (char [6])
SEQUENCE_NUMBER: The sequence number of the message
TRANSACTION_ID: The transaction id specified in the corresponding PMS Pending transaction command.
ID: The unique ID assigned by the NSE to the transaction. Note that this ID is assigned and returned when the transaction is created via PMS Pending transaction command.
ERROR_CODE:

- 0 – No error occurred
- 1 – Reserved
- 2 – The transaction failed due to repeated NAKs
- 3 – The transaction failed due to a timeout
- 4 – The transaction's data (specified in the PMS Pending transaction command) matched a configured PMS Redirector filter and was therefore dropped.

COMPLETION_CODE:

- 0 – Additional responses are anticipated for this transaction
- 1 – This is the final response to this transaction

DATA: The application data received (if any) from the attached PMS system that caused this PMS Transaction response to be generated.

Sample PMS Transaction Response Message:

```
<USG COMMAND="PMS_TRANSACTION_RESPONSE" VERSION="1.0" >
    <NSE_ID>112233</NSE_ID>
    <SEQUENCE_NUMBER>4871</SEQUENCE_NUMBER>
    <TRANSACTION_ID>123445</TRANSACTION_ID>
    <ID>233</ID >
    < ERROR_CODE>0</ERROR_CODE>
    < COMPLETION_CODE>1</COMPLETION_CODE>
    <DATA>DE|DA110715|TI090102|</DATA>
</USG>
```

7.3 PMS Unsolicited Response

Upon receipt of an application message from the attached PMS system that is not part of a transaction (i.e. response to a PMS Pending transaction command) a PMS Unsolicited response will be generated and sent to the Portal Page web server.

The DTD for this message is the following:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines PMS Unsolicited Response sent from NSE
-->

<!ELEMENT NSE_ID (#PCDATA)>
<!ELEMENT SEQUENCE_NUMBER (#PCDATA)>
<!ELEMENT DATA (#PCDATA)>

<!ELEMENT USG (NSE_ID, SEQUENCE_NUMBER, DATA)>
<!ATTLIST USG
   COMMAND CDATA #REQUIRED
   VERSION CDATA #REQUIRED
>
```

COMMAND attribute: 'PMS_UNSOLICITED_RESPONSE'
VERSION attribute: '1.0'
NSE_ID: ID of the NSE (char [6])
SEQUENCE_NUMBER: The sequence number of the message
DATA: The application data received from the attached PMS system that caused this PMS Unsolicited response to be generated.

Sample PMS Unsolicited Response Message:

```
<USG COMMAND="PMS_UNSOLICITED_RESPONSE" VERSION="1.0" >
    <NSE_ID>112233</NSE_ID>
    <SEQUENCE_NUMBER>4871</SEQUENCE_NUMBER>
    <DATA>DE|DA110715|TI090102|</DATA>
</USG>
```

8. WAN Status

The NSE sends this message to the Portal Page web server upon booting and whenever the primary wan interface of the NSE changes. Note that this message is only sent if the load balancing feature or the interface failover feature on the NSE is enabled. This is the XML Command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Wan Status Message sent from NSE
-->

<!ELEMENT PRIMARY_WAN_IP (#PCDATA)>

<!ELEMENT USG (PRIMARY_WAN_IP)>
<!ATTLIST USG
   COMMAND CDATA #REQUIRED
           ID CDATA #REQUIRED
>
```

Where:
COMMAND attribute: 'WAN_STATUS'
ID attribute: ID of the NSE (char [6])
PRIMARY_WAN_IP: IP address of the NSE'S PRIMARY WAN (char [18])

Sample command XML:

```
<USG COMMAND="WAN_STATUS" ID="0164b3">
        <PRIMARY_WAN_IP>192.168.100.102</PRIMARY_WAN_IP>
</USG>
```

9. XML Format for DAT table

The NSE will send the DAT table with this format after a get request is sent to the following Web address: **http[s]://NSE_IP/api/dat**. This is the XML Command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines DAT Table Message sent from the NSE
-->

<!ELEMENT SESS_NUM (#PCDATA)>
<!ELEMENT SUB_IP (#PCDATA)>
<!ELEMENT SUB_PORT (#PCDATA)>
<!ELEMENT SUB_MAC_ADDR (#PCDATA)>
<!ELEMENT SESS_NAT_IP (#PCDATA)>
<!ELEMENT SESS_NAT_PORT (#PCDATA)>
<!ELEMENT REMOTE_IP (#PCDATA)>
< !ELEMENT REMOTE_PORT (#PCDATA)>
<!ELEMENT SESS_PROTO (#PCDATA)>
<!ELEMENT SESS_STATE (#PCDATA)>
<!ELEMENT IDLE_TOUT (#PCDATA)>
<!ELEMENT SESS_TOUT (#PCDATA)>

<!ELEMENT NSE (SESSION_ENTRY)>
<!ELEMENT SESSION_ENTRY(SESS_NUM, SUB_IP, SUB_PORT, SUB_MAC_ADDR,
SESS_NAT_IP, SESS_NAT_PORT, REMOTE_IP, REMOTE_PORT, SESS_PROTO, SESS_STATE,
IDLE_TOUT, SESS_TOUT)>
<!ATTLIST NSE COMMAND CDATA #REQUIRED>
```

Where:
COMMAND attribute: ‘DAT_TABLE_RSP’
SESS_NUM: The Session Number in the Table
SUB_IP: The Subscriber IP address for this Session
SUB_PORT: The Subscribers Source Port for this Session
SUB_MAC_ADDR: The MAC address for the Subscriber for this Session.
SESS_NAT_IP: The IP address that this session has been translated to usually the NSE IP but sometimes an INAT address.
SESS_NAT_PORT: The source port from the NSE that this session is using.
REMOTE_IP: The destination IP for this Session.
REMOTE_PORT: The destination port for this translated session.
SESS_PROTO: The protocol that is being used in this session. (Usually TCP or UDP, ANY means it is an INAT session)
SESS_STATE: The State that the Session is in. (i.e. Established, Time_Wait, UDP_MAPPED, etc.)
IDLE_TOUT: The Idle timeout for this session.
SESS_TOUT: How long the session has been timing out.

Sample command XML:

```
<NSE COMMAND="DAT_TABLE_RSP">
    <SESSION_ENTRY>
     <SESS_NUM>1</SESS_NUM>
     <SUB_IP>10.0.0.13</SUB_IP>
     <SUB_PORT>1387</SUB_PORT>
     <SUB_MAC_ADDR>00:11:22:33:44:55</SUB_MAC_ADDR>
      <SESS_NAT_IP>192.168.100.102</SESS_NAT_IP>
      <SESS_NAT_PORT>5026</SESS_NAT_PORT>
      <REMOTE_IP>80.239.235.200</REMOTE_IP>
      <REMOTE_PORT>443</REMOTE_PORT>
      <SESS_PROTO>TCP</SESS_PROTO>
      <SESS_STATE>ESTABLISHED</SESS_STATE>
      <IDLE_TOUT>7</IDLE_TOUT>
     <SESS_TOUT>1793</SESS_TOUT>
   </SESSION_ENTRY>
  </NSE>
```

10. XML Format for Current Subscriber Table

The NSE will send the Current Subscriber table with this format after a get request is sent to the following Web address: **http[s]://NSE_IP/api/current**. This is the XML Command with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines Current Table Message sent from the NSE
-->

<!ELEMENT SUB_MAC_ADDR (#PCDATA)>
<!ELEMENT SUB_IP (#PCDATA)>
<!ELEMENT LOCATION (#PCDATA)>
<!ELEMENT ROOM_NUM (#PCDATA)>
<!ELEMENT USERNAME (#CDATA)>
<!ELEMENT GROUP_BW_ID (#PCDATA)>
<!ELEMENT BW_UP (#PCDATA)>
<!ELEMENT BW_DOWN (#PCDATA)>
<!ELEMENT THRU_UP (#PCDATA)>
<!ELEMENT THRU_DOWN (#PCDATA)>
<!ELEMENT SUB_AAA_STATE (#PCDATA)>
<!ELEMENT EXPIRY_TIME (#PCDATA)>
<!ELEMENT SUB_IDLE_TO (#PCDATA)>
<!ELEMENT BYTES_TX (#PCDATA)>
<!ELEMENT BYTES_RX (#PCDATA)>
<!ELEMENT PROXY_STATE (#PCDATA)>
<!ELEMENT NAT_IP (#PCDATA)>
<!ELEMENT CLASS(#PCDATA)>
<!ELEMENT IP_TYPE (#PCDATA)>
<!ELEMENT SMTP_REDIRECTION   (#PCDATA)>



<!ELEMENT NSE (SUBSCRIBER*)>
<!ELEMENT SUBSCRIBER (SUB_MAC_ADDR, SUB_IP, LOCATION, ROOM_NUM, USERNAME,
BW_UP, BW_DOWN, THRU_UP, THRU_DOWN, SUB_AAA_STATE, EXPIRY_TIME,
SUB_IDLE_TO, BYTES_TX, BYTES_RX, PROXY_STATE, NAT_IP, CLASS?
IP_TYPE?, SMPT_REDIRECTION)>
<!ATTLIST NSE COMMAND CDATA #FIXED “CURR_USERS_RSP”>
<!ATTLIST CLASS AVAILABLE CDATA #REQUIRED>
```

Where:
COMMAND attribute: ‘CURR_USERS_RSP’
SUB_MAC_ADDR: MAC Address of the Subscriber in the Table
SUB_IP: IP address of the Subscriber in the Table
LOCATION: The Port that the Subscriber is connected on for either VLAN or SNMP Query return.

ROOM_NUM: The Room Number that Matches the LOCATION information from the Port Location Table.
USERNAME: Subscribers Username
GROUP_BW_ID: The Group Bandwidth Policy number.
BW_UP: The Configured Maximum Upstream Bandwidth for this Subscriber
BW_DOWN: The Configured Maximum Downstream Bandwidth for this Subscriber
THRU_UP: The Current amount of upstream throughput this subscriber is utilizing.
THRU_DOWN: The Current amount of downstream throughput this subscriber is utilizing.
SUB_AAA_STATE: The State of the Subscriber (i.e. Valid, Pending, Valid-Radius, etc.)
EXPIRY_TIME: The amount of time left before the subscriber session times out.
SUB_IDLE_TO: The amount of idle time left before the subscriber is removed from the current subscriber list.
BYTES_TX: Number of Bytes sent by the subscriber.
BYTES_RX: Number of Bytes sent to the subscriber.
PROXY_STATE: Current Proxy State of the Subscriber
NAT_IP: The NAT IP address that is used for data flows between this subscriber and the network (will show as 0.0.0.0 if no NAT IP address has been assigned to the subscriber yet).
CLASS: The CBQ Class that the subscriber is assigned to. The "AVAILABLE" attribute of this element indicates whether the indicated class is available on the interface the subscriber is currently assigned to.
IP_TYPE (optional): If the subscriber has been assigned an address by the NSE, this parameter indicates whether the address was assigned from a public ("PUBLIC") or private ("PRIVATE") DHCP POOL. Note: This parameter indicates the IP_TYPE at the time of address assignment. Subsequent changes to DHCP Pools have no effect on this parameter.
SMTP_REDIRECTION: String indicating if the SMTP protocol redirection is enabled for this subscriber ("ENABLED" or "DISABLED").

Sample response XML:

```
<NSE COMMAND="CURR_USERS_RSP">
<SUBSCRIBER>
 <SUB_MAC_ADDR>00:11:22:33:44:55</SUB_MAC_ADDR>
 <SUB_IP>192.168.100.102</SUB_IP>
 <LOCATION>0</LOCATION>
 <ROOM_NUM />
 <USER_NAME>
 <![CDATA[ "test"  ]]>
</USER_NAME>
<GROUP_BW_ID>1</ GROUP_BW_ID >
```

<BW_UP>**0**</BW_UP>
<BW_DOWN>**0**</BW_DOWN>
<THRU_UP>**0-0**</THRU_UP>
<THRU_DOWN>**0-0**</THRU_DOWN>
<SUB_AAA_STATE>**Valid**</SUB_AAA_STATE>
<EXPIRY_TIME>**3 hrs 52 min**</EXPIRY_TIME>
<SUB_IDLE_TO>**20 mins : 0 sec**</SUB_IDLE_TO>
<BYTES_TX>**11708**</BYTES_TX>
<BYTES_RX>**10111**</BYTES_RX>
<PROXY_STATE>**Off**</PROXY_STATE>
<NAT_IP>**67.130.148.131**</NAT_IP>
<CLASS AVAILABLE=”TRUE” >**a**</CLASS>
<IP_TYPE >**PRIVATE**</IP_TYPE>
<SMTP_REDIRECTION >**DISABLED**</SMTP_REDIRECTION>
</SUBSCRIBER>
</NSE>

11. XML Format for Subscribers in Authorized Database

The list of subscriber records stored in the authorized database is obtained by sending an HTTP GET request to the following URL: **http[s]://NSE_IP/api/subAuth**

Response for the HTTP GET *subAuth*

The response is an HTTP reply containing XML data with the following DTD:

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
  DTD defines response for the HTTP GET subAuth request
-->

<!ELEMENT MAC_ADDR   (#PCDATA)>
<!ELEMENT USER_NAME (#CDATA)>
<!ELEMENT IP_ADDR      (#PCDATA)>
<!ELEMENT SUBNET      (#PCDATA)>
<!ELEMENT EXPIRY_TIME_SECS (#PCDATA)>
<!ELEMENT EXPIRED (#PCDATA)>
<!ELEMENT AMT_PAID     (#PCDATA)>
<!ELEMENT AMT_LEFT     (#PCDATA)>
<!ELEMENT USER_DEF1 (#CDATA)>
<!ELEMENT USER_DEF2 (#CDATA)>
<!ELEMENT AUTH_METHOD (#PCDATA)>
<!ELEMENT COUNT_DOWN (#PCDATA)>
<!ELEMENT COUNTING_DOWN (#PCDATA)>
<!ELEMENT IP_TYPE (#PCDATA)>
<!ELEMENT MAX_BW_UP  (#PCDATA)>
<!ELEMENT MAX_BW_DOWN (#PCDATA)>
<!ELEMENT BILLING_PLAN (#PCDATA)>
<!ELEMENT QOS_POLICY   (#PCDATA)>
<!ELEMENT CLASS (#PCDATA)>
<!ELEMENT SMTP_REDIRECTION   (#PCDATA)>

<!ELEMENT SUBSCRIBER_AUTH (MAC_ADDR, USER_NAME, USER_DEF1, USER_DEF2,
IP_ADDR, SUBNET, EXPIRY_TIME_SECS, EXPIRED, AMT_PAID, AMT_LEFT, AUTH_METHOD,
COUNT_DOWN, COUNTING_DOWN, IP_TYPE?, MAX_BW_UP?, MAX_BW_DOWN?,
BILLING_PLAN,  QOS_POLICY?, CLASS, SMTP_REDIRECTION)>
<!ELEMENT USG (SUBSCRIBER_AUTH*)>

<!ATTLIST USG
COMMAND CDATA #FIXED “SUBSCRIBER_AUTH”
>
```

Where:
COMMAND attribute: ‘SUBSCRIBER_AUTH’
MAC_ADDR: Subscriber’s MAC address, exactly 12 hex-ascii characters in length. An empty string is returned if subscriber was added by name.

USER_NAME: Subscriber's username, up to 96 hex-ascii characters. An empty string is returned if subscriber was added by MAC.
IP_ADDR: Subscriber's IP address, up to 15 characters in length. (May not reflect the correct IP address assigned to this subscriber.) This value may change at the IP update time.
SUBNET: Subscriber's subnet.
EXPIRY_TIME_SECS: The amount of time left, in seconds, before the subscriber account times out. An empty string will be returned if there is no expiration time or this subscriber has expired.
EXPIRED: String indicating if this subscriber has expired. ("TRUE" or "FALSE").
AMT_PAID: Amount paid by the user of this account.
AMT_LEFT: Amount left on this account. For x-over-y subscribers this value does not reflect the The actual amount left on this account, which will be updated at the logout time.
USER_DEF1: User definable string (char [128]).
USER_DEF2: User definable string (char [128]).
AUTH_METHOD: String indicating by what method the subscriber was added to the authorized persistent database. Values are: "PMS", "CREDIT_CARD", "XML", "ADMIN". Radius and post-paid PMS subscribers will not appear in the authorized database. Some other methods of authorization may be added in the future and the users of this command should be prepared to handle such cases.
Note: "XML" will be returned for subscribers that were added via XML commands, regardless of the payment method.

COUNT_DOWN: String indicating if Count-down starts after Login for this subscriber; ("ENABLED" or "DISABLED").
COUNTING_DOWN: String indicating if the time is running down for this subscriber. ("TRUE" or "FALSE").
IP_TYPE (optional): String indicating what kind of IP the user is authorized to use. ("PRIVATE" or "PUBLIC").
MAX_BW_UP (optional): Configured maximum upstream bandwidth, in Kbps, for this subscriber. An empty string will be returned if this parameter was not configured when the account was created – meaning UNLIMITED.
MAX_BW_DOWN (optional): Configured maximum downstream bandwidth, in Kbps, for this subscriber. An empty string will be returned if this parameter was not configured when the account was created – meaning UNLIMITED.
BILLING_PLAN: Plan number associated with this account. An empty string is returned if there is no associated plan for this subscriber.
QOS_POLICY (optional): QoS policy associated with this account, up to 16 characters in length. An empty string will be returned if no policy is assigned to this subscriber.
CLASS: The CBQ class that has been assigned to the subscriber.
SMTP_REDIRECTION: String indicating if the SMTP protocol redirection is enabled for this subscriber. ("ENABLED", or "DISABLED")
Note: This does not take into account a global status of SMTP redirect. Rather, how the individual subscriber was configured.

Note about optional elements:

Elements specified as optional will not be present if they are not licensed on a particular NSE or if they are not implemented on a hardware platform on which the NSE is running. However, if the feature is licensed but was not configured for the particular subscriber, the element will be present in the response but will contain no data. For example, if the quality of service on a particular unit is licensed but user did not select policy during configuration, the element QOS_POLICY will be present but will contain en empty string.

Implementation Notes for Portal/EWS Developers:

1) **Must gracefully ignore elements not recognized:** In the future, as new NSE features are implemented or as new requirements arise for the *subscriber_query_auth* command, new elements may be added to the response. An implementation must be prepared to gracefully ignore any unrecognized elements it may receive.
2) **Must gracefully handle missing optional elements:** Elements specified as optional in the DTD may or may not be present. An implementation must handle either case gracefully. See "note about optional elements" above for more detail.
3) **Must gracefully handle empty elements:** Many of the elements may be present but be empty of data, depending on NSE configuration and subscriber state. An implementation must be prepared to handle empty elements gracefully. See detailed element descriptions above and the "note about optional elements" above for more details.

Sample Response XML:

```
<USG COMMAND="SUBSCRIBER_AUTH">
  <SUBSCRIBER_AUTH>
    <MAC_ADDR>001122334455</MAC_ADDR>
    <USER_NAME><![CDATA[Gonzales]]></USER_NAME>
    <IP_ADDR>10.0.0.12</IP_ADDR>
    <SUBNET></SUBNET>
    <EXPIRY_TIME_SECS>40809</EXPIRY_TIME_SECS>
    <AMT_PAID>678.55</AMT_PAID>
    <AMT_LEFT>16.35</AMT_LEFT>
    <USER_DEF1><![CDATA[meeting room1]]></USER_DEF1>
    <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
    <AUTH_METHOD>ADMIN</AUTH_METHOD>
    <COUNT_DOWN>ENABLED</COUNT_DOWN>
    <COUNTING_DOWN>TRUE</COUNTING_DOWN>
    <IP_TYPE>PRIVATE</IP_TYPE>
    <MAX_BW_UP>512</MAX_BW_UP>
    <MAX_BW_DOWN>1024</MAX_BW_DOWN>
    <BILLING_PLAN>5</BILLING_PLAN>
    <QOS_POLICY>RH_102</QOS_POLICY>
    <CLASS>a</CLASS>
    <SMTP_REDIRECTION>ENABLED</SMTP_REDIRECTION>
  </SUBSCRIBER_AUTH>
  …
  <SUBSCRIBER_AUTH>
    <MAC_ADDR>001122334456</MAC_ADDR>
    <USER_NAME><![CDATA[Johnson]]></USER_NAME>
    <IP_ADDR>67.130.130.12</IP_ADDR>
    <SUBNET></SUBNET>
    <EXPIRY_TIME_SECS>2400</EXPIRY_TIME_SECS>
    <AMT_PAID>67.55</AMT_PAID>
    <AMT_LEFT>1.35</AMT_LEFT>
    <USER_DEF1><![CDATA[meeting room1]]></USER_DEF1>
```

```
        <USER_DEF2><![CDATA[whatever string]]></USER_DEF2>
        <AUTH_METHOD>XML</AUTH_METHOD>
        <COUNT_DOWN>ENABLED</COUNT_DOWN>
        <COUNTING_DOWN>FALSE</COUNTING_DOWN>
        <IP_TYPE>PUBLIC</IP_TYPE>
        <MAX_BW_UP>512</MAX_BW_UP>
        <MAX_BW_DOWN>1024</MAX_BW_DOWN>
        <BILLING_PLAN>0</BILLING_PLAN>
        <QOS_POLICY></QOS_POLICY>
        <CLASS>b</CLASS>
        <SMTP_REDIRECTION>ENABLED</SMTP_REDIRECTION>
    </SUBSCRIBER_AUTH>
</USG>
```

The status of the response will be conveyed through the standard http protocol mechanism.

12. XML Format for Group Bandwidth Policy List

The NSE will send an XML-encoded list representing installed bandwidth policies after a get request is sent to the following Web address: **http[s]://NSE_URI/api/bw/v1/groupPolicy**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (GROUP_BW_POL*)>
<!ATTLIST USG COMMAND CDATA #FIXED “GROUP_BW_POLICIES”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT GROUP_BW_POL (ID, MAX_BW_UP, MAX_BW_DOWN)>

<!-- ID contains an unsigned integer number in base 10.  It represents a unique identifier of a policy.  Valid range is between 1 and 16777215 inclusively -->
<!ELEMENT ID (#PCDATA)>

<!-- MAX_BW_UP contains an unsigned integer number in base 10. It represents the maximum upstream (ie towards the Internet) bandwidth of the policy. -->
<!ELEMENT MAX_BW_UP (#PCDATA)>

<!-- MAX_BW_DOWN contains an unsigned integer number in base 10. It represents the maximum downstream bandwidth of the policy. -->
<!ELEMENT MAX_BW_DOWN (#PCDATA)>
```

Note: The DTD for this command is stored in a file called “GroupBwPolicies-1.0.dtd” that can be accessed on the Nomadix web site.

Sample response XML:

```
<USG COMMAND=”GROUP_BW_POLICIES” VERSION="1.0">
  <GROUP_BW_POL>
    <ID>10</ID>
    <MAX_BW_UP>1024</MAX_BW_UP>
    <MAX_BW_DOWN>2048</MAX_BW_DOWN>
  </GROUP_BW_POL>
  <GROUP_BW_POL>
    <ID>17</ID>
    <MAX_BW_UP>4096</MAX_BW_UP>
    <MAX_BW_DOWN>4096</MAX_BW_DOWN>
  </GROUP_BW_POL>
</USG>
```

13. XML Format for Individual Group Bandwidth Policy

The NSE will send an XML-encoded representation of a bandwidth policy after a get request is sent to the following Web address: **http[s]://NSE_URI/api/bw/v1/groupPolicy/policyNumber**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (GROUP_BW_POL)>
<!ATTLIST USG COMMAND CDATA #FIXED “GROUP_BW_POLICY”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT GROUP_BW_POL (ID, MAX_BW_UP, MAX_BW_DOWN)>

<!-- ID contains an unsigned integer number in base 10.  It represents a unique identifier of a policy.  Valid range is between 1 and 16777215 inclusively -->
<!ELEMENT ID (#PCDATA)>

<!-- MAX_BW_UP contains an unsigned integer number in base 10. It represents the maximum upstream (ie towards the Internet) bandwidth of the policy. -->
<!ELEMENT MAX_BW_UP (#PCDATA)>

<!-- MAX_BW_DOWN contains an unsigned integer number in base 10. It represents the maximum downstream bandwidth of the policy. -->
<!ELEMENT MAX_BW_DOWN (#PCDATA)>
```

Note: The DTD for this command is stored in a file called “GroupBwPolicy-1.0.dtd” that can be accessed on the Nomadix web site.

Sample response XML:

```
<USG COMMAND=”GROUP_BW_POLICY” VERSION="1.0">
  <GROUP_BW_POL>
    <ID>10</ID>
    <MAX_BW_UP>1024</MAX_BW_UP>
    <MAX_BW_DOWN>2048</MAX_BW_DOWN>
  </GROUP_BW_POL>
</USG>
```

14. XML Format for PMS Pending Transaction List

The NSE will send an XML-encoded list representing PMS Pending Transactions after a get request is sent to the following Web address: **http[s]://NSE_URI/api/pmsRedirection/v1/pendingTransaction**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (P_TRANSACTION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PMS_PENDING_TRANSACTIONS”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT P_TRANSACTION (ID, LINK_STATE, TRANSACTION_ID, DATA)>

<!-- ID contains an unsigned integer number in base 10.  It represents a unique identifier of a transaction.
Valid range is between 1 and 16777215 inclusively -->
<!ELEMENT ID (#PCDATA)>

<!—LINK_STATE is present only if the serial link between the NSE and the attached PMS device is
down.  In this case it contains the value “DOWN”. -->
<!ELEMENT LINK_STATE (#PCDATA)>

<!—TRANSACTION_ID contains an unsigned integer number in base 10. It contains the transaction id
that was specified when the transaction when the transaction was created, or 0 if a transaction id was not
specified. . -->
<!ELEMENT TRANSACTION_ID (#PCDATA)>

<!—DATA contains the data that will be sent to the attached PMS system. Before sending, the data is
framed with an ETX (hex 02) and an STX (hex 03) and appended with a checksum-->
 <!ELEMENT DATA (#PCDATA)>
```

Sample response XML:

```
<USG COMMAND=”PMS_PENDING_TRANSACTIONS” VERSION="1.0">
   <P_TRANSACTION URI=”/api/pmsRedirector/v1/pendingTransaction/2>
      <ID>2</ID>
      <LINK_STATE>DOWN</LINK_STATE>
      <TRANSACTION_ID>111111</TRANSACTION_ID>
      <DATA>
       PS|RN1002  |PTC|TA1100|S11000|T1100|DA110810|TI113143|P#0005|CTPlan A = Pri
       </DATA>
```

```
    </P_TRANSACTION>
   <P_TRANSACTION URI=”/api/pmsRedirector/v1/pendingTransaction/3>
      <ID>3</ID>
      <LINK_STATE>DOWN</LINK_STATE>
      <TRANSACTION_ID>223344</TRANSACTION_ID>
      <DATA>
       PS|RN1015  |PTC|TA2200|S12000|T1200|DA110810|TI113147|P#0005|CTPlan A = Pri
       </DATA>
    </P_TRANSACTION>
</USG>
```

15. XML Format for Individual PMS Pending Transaction

The NSE will send an XML-encoded representation of a PMS Pending Transaction after a get request is sent to the following Web address: **http[s]://NSE_URI/api/pmsRedirection/v1/pendingTransaction/job id**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (P_TRANSACTION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PMS_PENDING_TRANSACTION”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT P_TRANSACTION (ID, LINK_STATE, TRANSACTION_ID, DATA)>

<!-- ID contains an unsigned integer number in base 10.  It represents a unique identifier of a transaction.
Valid range is between 1 and 16777215 inclusively -->
<!ELEMENT ID (#PCDATA)>

<!—LINK_STATE is present only if the serial link between the NSE and the attached PMS device is
down.  In this case it contains the value “DOWN”. -->
<!ELEMENT LINK_STATE (#PCDATA)>

<!—TRANSACTION_ID contains an unsigned integer number in base 10. It contains the transaction id
that was specified when the transaction when the transaction was created, or 0 if a transaction id was not
specified. . -->
<!ELEMENT TRANSACTION_ID (#PCDATA)>

<!—DATA contains the data that will be sent to the PMS.  Before sending, the data is framed in an EXT
and STX characters and appended with a checksum. . -->
<!ELEMENT DATA (#PCDATA)>
```

Sample response XML:

```
<USG COMMAND=”PMS_PENDING_TRANSACTION” VERSION="1.0">
  <P_TRANSACTION URI=”/api/pmsRedirector/v1/pendingTransaction/2>
    <ID>2</ID>
    <LINK_STATE>DOWN</LINK_STATE>
    <TRANSACTION_ID>111111</TRANSACTION_ID>
    <DATA>
     PS|RN1002  |PTC|TA1100|S11000|T1100|DA110810|TI113143|P#0005|CTPlan A = Pri
     </DATA>
  </P_TRANSACTION>
```

16. XML Format for PMS Completed Transaction List

The NSE will send an XML-encoded list representing PMS Completed Transactions after a get request is sent to the following Web address: **http[s]://NSE_URI/api/pmsRedirection/v1/completedTransaction**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (C_TRANSACTION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PMS_COMPLETED_TRANSACTIONS”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT C_TRANSACTION (ID, TRANSACTION_ID, COMPLETION_STATUS,
RESPONSE_COUNT)>

<!-- ID contains an unsigned integer number in base 10.  It represents a unique identifier of a transaction.
Valid range is between 1 and 16777215 inclusively -->
<!ELEMENT ID (#PCDATA)>

<!—TRANSACTION_ID contains an unsigned integer number in base 10. It contains the transaction id
that was specified when the transaction when the transaction was created, or 0 if a transaction id was not
specified. -->
<!ELEMENT TRANSACTION_ID (#PCDATA)>

<!—COMPLETION_STATUS indicates the status of the transaction and is one of the following values:
  0 – The transaction is complete.
  1 – The transaction is not yet complete, additional responses from the attached PMS device are expected.
  2 – The transaction was “NAKed” by the attached PMS device and therefore failed.
  3 – The transaction timed out.  A response to the transaction was not received.
  4 – The transaction was filtered by the NSE and not transmitted to the attached PMS device. -->
<!ELEMENT COMPLETION_STATUS (#PCDATA)>

<!—RESPONSE_COUNT is an unsigned number in base 10.  It represents the number of responses that
were received from the PMS that were considered responses to the transaction. -->
<!ELEMENT RESPONSE_COUNT (#PCDATA)>
```

Sample response XML:

```
<USG COMMAND=”PMS_COMPLETED_TRANSACTIONS” VERSION="1.0">
   <C_TRANSACTION URI=”/api/pmsRedirector/v1/completedTransaction/2>
      <ID>2</ID>
      <TRANSACTION_ID>111111</TRANSACTION_ID>
      <COMPLETION_STATUS>0</COMPLETION_STATUS >
      <RESPONSE_COUNT>1</RESPONSE_COUNT >
   </C_TRANSACTION>
```

```
  <C_TRANSACTION URI=”/api/pmsRedirector/v1/completedTransaction/3>
    <ID>3</ID>
    <TRANSACTION_ID>223344</TRANSACTION_ID>
    <COMPLETION_STATUS>0</COMPLETION_STATUS >
    <RESPONSE_COUNT>1</RESPONSE_COUNT >
   </C_TRANSACTION>
</USG>
```

17. XML Format for Individual PMS Completed Transaction

The NSE will send an XML-encoded representation of a PMS Completed Transaction after a get request is sent to the following Web address: **http[s]://NSE_URI/api/pmsRedirection/v1/completedTransaction/job id**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (C_TRANSACTION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PMS_COMPLETED_TRANSACTION”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT C_TRANSACTION (ID, TRANSACTION_ID, COMPLETION_STATUS,
RESPONSE_COUNT)>

<!-- ID contains an unsigned integer number in base 10.  It represents a unique identifier of a transaction.
Valid range is between 1 and 16777215 inclusively -->
<!ELEMENT ID (#PCDATA)>

<!—TRANSACTION_ID contains an unsigned integer number in base 10. It contains the transaction id
that was specified when the transaction when the transaction was created, or 0 if a transaction id was not
specified. -->
<!ELEMENT TRANSACTION_ID (#PCDATA)>

<!—COMPLETION_STATUS indicates the status of the transaction and is one of the following values:
  0 – The transaction is complete.
  1 – The transaction is not yet complete, additional responses from the attached PMS device are expected.
  2 – The transaction was “NAKed” by the attached PMS device and therefore failed.
  3 – The transaction timed out.  A response to the transaction was not received.
  4 – The transaction was filtered by the NSE and not transmitted to the attached PMS device. -->
<!ELEMENT COMPLETION_STATUS (#PCDATA)>

<!—RESPONSE_COUNT is an unsigned number in base 10.  It represents the number of responses that
were received from the PMS that were considered responses to the transaction. -->
<!ELEMENT RESPONSE_COUNT (#PCDATA)>
```

Sample response XML:

```
<USG COMMAND=”PMS_COMPLETED_TRANSACTION” VERSION="1.0">
   <C_TRANSACTION URI=”/api/pmsRedirector/v1/completedTransaction/2>
      <ID>2</ID>
      <TRANSACTION_ID>111111</TRANSACTION_ID>
      <COMPLETION_STATUS>0</ COMPLETION_STATUS >
      <RESPONSE_COUNT>1</ RESPONSE_COUNT >
   </C_TRANSACTION>
```

18. XML Format for Port-location List

The NSE will send an XML-encoded list representing Port-locations after a get request is sent to the following Web address: **http[s]://NSE_URI/api/portLocation/v1/portLocations**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (P_LOCATION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PORT_LOCATIONS”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT P_LOCATION (PORT, LOCATION, DESCRIPTION, SUBNET, MODEM_MAC, STATE,
ALLOW_RADIUS, ALLOW_PMS, ALLOW_CREDIT, PLAN_NUMBER, QOS_POLICY,
INTRA_PORT,ALLOW_DHCPC,FACEBOOK_LOGIN)>

<!-- PORT contains an unsigned integer number in base 10. It contains the port that was specified when the Port-
location was created. -->
<!ELEMENT PORT (#PCDATA)>

<!-- LOCATION indicates the name of the location -->
<!ELEMENT LOCATION (#CDATA)>

<!-- DESCRIPTION indicates a description of the location -->
<!ELEMENT DESCRIPTION (#CDATA)>

<!-- SUBNET indicates a subnet assigned to the location: must be either 0.0.0.0 or one of the subnets of a configured
DHCP pool -->
<!ELEMENT SUBNET (#PCDATA)>

<!-- MODEM_MAC indicates the MAC address of a RiverDelta or Elastic Networks concentrator for the location -->
<!ELEMENT MODEM_MAC (#PCDATA)>

<!-- STATE indicates the charging method used for the location. It is one of the following values:
 0 – No charge.
 1 – Charge.
 2 – Blocked. -->
<!ELEMENT STATE (#PCDATA)>

<!-- ALLOW_RADIUS indicates whether a subscriber can be charged with RADIUS (true or false) -->
<!ELEMENT ALLOW_RADIUS (#PCDATA)>

<!-- ALLOW_PMS indicates whether a subscriber can be charged with PMS (true or false) -->
<!ELEMENT ALLOW_PMS (#PCDATA)>

<!-- ALLOW_CREDIT indicates whether a subscriber can be charged with Credit Card (true or false) -->
<!ELEMENT ALLOW_CREDIT (#PCDATA)>

<!-- PLAN_NUMBER indicates which billing plans are available for the location. It is one of the following values:
 -1 – All plans.
 0 to 5 – A specific plan identified by its number -->
<!ELEMENT PLAN_NUMBER (#PCDATA)>

<!-- QOS_POLICY indicates a default QoS policy name for the location -->
<!ELEMENT ALLOW QOS_POLICY (#PCDATA)>

<!-- INTRA_PORT indicates whether Intra-port communication is enabled (true or false) -->
<!ELEMENT INTRA_PORT (#PCDATA)>
```

```
<!—ALLOW_DHCPC  indicates whether DHCP client messages will be processed (true or false) -->
<!ELEMENT ALLOW_DHCPC (#PCDATA)>

<!—FACEBOOK_LOGIN  indicates whether Facebook logins will be allowed (true or false) -->
<!ELEMENT FACEBOOK_LOGIN (#PCDATA)>
```

Sample response XML:

```
<USG COMMAND=”PORT_LOCATIONS” VERSION="1.0">
   <P_LOCATION URI=”/api/portLocation/v1/portLocations/1>
      <ID>1</ID>
      <PORT>10</PORT>
      <LOCATION>room 10</LOCATION>
      <SUBNET>0.0.0.0</SUBNET>
      <MODEM_MAC>00:00:00:00:00:00</MODEM_MAC>
      <STATE>0</STATE>
      <ALLOW_RADIUS>false</ALLOW_RADIUS>
      <ALLOW_PMS>false</ALLOW_PMS>
      <ALLOW_CREDIT>false</ALLOW_CREDIT>
      <PLAN_NUMBER>-1</PLAN_NUMBER>
      <QOS_POLICY>silver</QOS_POLICY>
      <INTRA_PORT>false</INTRA_PORT>
      <ALLOW_DHCPC>true</ALLOW_DHCPC>
      <FACEBOOK_LOGIN>false</FACEBOOK_LOGIN>
  </P_LOCATION>
</USG>
```

19. XML Format for Getting Individual Port-location

The NSE will send an XML-encoded representation of a Port-location after a get request is sent to the following Web address: **http[s]://NSE_URI/api/portLocation/v1/portLocations/location id**. This is the XML Command with the following DTD:

```
<!ELEMENT USG (P_LOCATION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PORT_LOCATION”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT P_LOCATION (PORT, LOCATION, DESCRIPTION, SUBNET, MODEM_MAC, STATE,
ALLOW_RADIUS, ALLOW_PMS, ALLOW_CREDIT, PLAN_NUMBER, QOS_POLICY,
INTRA_PORT,ALLOW_DHCPC,FACEBOOK_LOGIN)>

<!-- PORT contains an unsigned integer number in base 10. It contains the port that was specified when the Port-
location was created. -->
<!ELEMENT PORT (#PCDATA)>

<!-- LOCATION indicates the name of the location -->
<!ELEMENT LOCATION (#CDATA)>

<!-- DESCRIPTION indicates a description of the location -->
<!ELEMENT DESCRIPTION (#CDATA)>

<!-- SUBNET indicates a subnet assigned to the location: must be either 0.0.0.0 or one of the subnets of a configured
DHCP pool -->
<!ELEMENT SUBNET (#PCDATA)>

<!-- MODEM_MAC indicates the MAC address of a RiverDelta or Elastic Networks concentrator for the location -->
<!ELEMENT MODEM_MAC (#PCDATA)>

<!-- STATE indicates the charging method used for the location. It is one of the following values:
 0 – No charge.
 1 – Charge.
 2 – Blocked. -->
<!ELEMENT STATE (#PCDATA)>

<!-- ALLOW_RADIUS indicates whether a subscriber can be charged with RADIUS (true or false) -->
<!ELEMENT ALLOW_RADIUS (#PCDATA)>

<!-- ALLOW_PMS indicates whether a subscriber can be charged with PMS (true or false) -->
<!ELEMENT ALLOW_PMS (#PCDATA)>

<!-- ALLOW_CREDIT indicates whether a subscriber can be charged with Credit Card (true or false) -->
<!ELEMENT ALLOW_CREDIT (#PCDATA)>

<!-- PLAN_NUMBER indicates which billing plans are available for the location. It is one of the following values:
 -1 – All plans.
 0 to 5 – A specific plan identified by its number -->
<!ELEMENT PLAN_NUMBER (#PCDATA)>

<!-- QOS_POLICY indicates a default QoS policy name for the location -->
<!ELEMENT QOS_POLICY (#PCDATA)>

<!-- INTRA_PORT indicates whether Intra-port communication is enabled (true or false) -->
<!ELEMENT INTRA_PORT (#PCDATA)>
```

```
<!—ALLOW_DHCPC  indicates whether DHCP client messages will be processed (true or false) -->
<!ELEMENT ALLOW_DHCPC (#PCDATA)>

<!—FACEBOOK_LOGIN  indicates whether Facebook logins will be allowed (true or false) -->
<!ELEMENT FACEBOOK_LOGIN (#PCDATA)>
```

Sample response XML:

```
<USG COMMAND=”PORT_LOCATION” VERSION="1.0">
   <P_LOCATION URI=”/api/portLocation/v1/portLocations/1>
      <ID>1</ID>
      <PORT>10</PORT>
      <LOCATION>room 10</LOCATION>
      <SUBNET>0.0.0.0</SUBNET>
      <MODEM_MAC>00:00:00:00:00:00</MODEM_MAC>
      <STATE>0</STATE>
      <ALLOW_RADIUS>false</ALLOW_RADIUS>
      <ALLOW_PMS>false</ALLOW_PMS>
      <ALLOW_CREDIT>false</ALLOW_CREDIT>
      <PLAN_NUMBER>-1</PLAN_NUMBER>
      <QOS_POLICY>silver</QOS_POLICY>
      <INTRA_PORT>false</INTRA_PORT>
      <ALLOW_DHCPC>true</ALLOW_DHCPC>
      <FACEBOOK_LOGIN>false</FACEBOOK_LOGIN>
   </P_LOCATION>
</USG>
```

20. XML Format for Setting Individual Port-location

NOTE: This command should be sent as a POST or PUT request to the following address:

http[s]://NSE_URI/api/portLocation/v1/portLocations[/location id].

In the case of a post request, the location id is omitted, as a new entry will be created in the table with a new location id.
Submit an XML command to be processed by the NSE. This is an XML Command with the following DTD:

```
<!ELEMENT USG (P_LOCATION)>
<!ATTLIST USG COMMAND CDATA #FIXED “PORT_LOCATION”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT P_LOCATION (PORT, LOCATION, DESCRIPTION, SUBNET, MODEM_MAC, STATE,
ALLOW_RADIUS, ALLOW_PMS, ALLOW_CREDIT, PLAN_NUMBER, QOS_POLICY,
INTRA_PORT,ALLOW_DHCPC,FACEBOOK_LOGIN)>

<!-- PORT contains an unsigned integer number in base 10. It contains the port that was specified when the Port-
location was created. -->
<!ELEMENT PORT (#PCDATA)>

<!-- LOCATION indicates the name of the location -->
<!ELEMENT LOCATION (#CDATA)>

<!-- DESCRIPTION indicates a description of the location -->
<!ELEMENT DESCRIPTION (#CDATA)>

<!-- SUBNET indicates a subnet assigned to the location: must be either 0.0.0.0 or one of the subnets of a configured
DHCP pool -->
<!ELEMENT SUBNET (#PCDATA)>

<!-- MODEM_MAC indicates the MAC address of a RiverDelta or Elastic Networks concentrator for the location -->
<!ELEMENT MODEM_MAC (#PCDATA)>

<!-- STATE indicates the charging method used for the location. It is one of the following values:
 0 – No charge.
 1 – Charge.
 2 – Blocked. -->
<!ELEMENT STATE (#PCDATA)>

<!-- ALLOW_RADIUS indicates whether a subscriber can be charged with RADIUS (true or false) -->
<!ELEMENT ALLOW_RADIUS (#PCDATA)>

<!-- ALLOW_PMS indicates whether a subscriber can be charged with PMS (true or false) -->
<!ELEMENT ALLOW_PMS (#PCDATA)>

<!-- ALLOW_CREDIT indicates whether a subscriber can be charged with Credit Card (true or false) -->
<!ELEMENT ALLOW_CREDIT (#PCDATA)>

<!-- PLAN_NUMBER indicates which billing plans are available for the location. It is one of the following values:
 -1 – All plans.
 0 to 5 – A specific plan identified by its number -->
<!ELEMENT PLAN_NUMBER (#PCDATA)>

<!-- QOS_POLICY indicates a default QoS policy name for the location -->
<!ELEMENT QOS_POLICY (#PCDATA)>

<!-- INTRA_PORT indicates whether Intra-port communication is enabled (true or false) -->
<!ELEMENT INTRA_PORT (#PCDATA)>
```

```
<!—ALLOW_DHCPC  indicates whether DHCP client messages will be processed (true or false) -->
<!ELEMENT ALLOW_DHCPC (#PCDATA)>

<!—FACEBOOK_LOGIN  indicates whether Facebook logins will be allowed (true or false) -->
<!ELEMENT FACEBOOK_LOGIN (#PCDATA)>
```

Sample command XML:

```
<USG COMMAND=”PORT_LOCATION” VERSION="1.0">
   <P_LOCATION URI=”/api/portLocation/v1/portLocations>
      <PORT>10</PORT>
      <LOCATION>room 10</LOCATION>
      <SUBNET>0.0.0.0</SUBNET>
      <MODEM_MAC>00:00:00:00:00:00</MODEM_MAC>
      <STATE>0</STATE>
      <ALLOW_RADIUS>false</ALLOW_RADIUS>
      <ALLOW_PMS>false</ALLOW_PMS>
      <ALLOW_CREDIT>false</ALLOW_CREDIT>
      <PLAN_NUMBER>-1</PLAN_NUMBER>
     <QOS_POLICY>silver</QOS_POLICY>
      <INTRA_PORT>false</INTRA_PORT>
      <ALLOW_DHCPC>true</ALLOW_DHCPC>
      <FACEBOOK_LOGIN>false</FACEBOOK_LOGIN>
   </P_LOCATION>
</USG>
```

21. XML Format for Removing Individual Port-location

NOTE: This command should be sent as a DELETE request to the following address:

http[s]://NSE_URI/api/portLocation/v1/portLocations/location id.

No XML command is required and no XML response is returned.

22. Active Routing Tables Commands

22.1 Listing Active Routes

Active routes can be listed via HTTP (s) "GET" requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/activeRoutes[/table id][/route id]**. If the optional table id is specified, only those routes for the specified table are listed. If, in addition to a table id, a route id is specified, the single specified route is listed.

Each active route returned will be XML-encoded as defined by the Active Route DTD (see section 22.5)

22.2 Deleting Active Routes

Active routes can be deleted via HTTP(s) "DELETE" requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/activeRoutes/table id/route id.** Note that only a single route can be deleted at a time. The route's table id and route id must be specified.

22.3 Table IDs

The system routing table has an ID of 254.

When the Load Balancing feature of the NSE is enabled, an additional routing table is created for each NSE Ethernet interface. These routing tables are used to route subscriber traffic. The routing table associated with the "WAN" labeled interface is used to route subscriber traffic for those subscribers assigned to WAN. Similarly, the routing table associated with the "Eth1" labeled interface is used to route subscriber traffic for those subscribers assigned to Eth1.

The IDs of these routing tables are: 1 for WAN, 2 for Eth1, 3 for Eth2, etc.

22.4 Route IDs

Each route has a unique id assigned by the system at the time of it's creation.

22.5 Active Route DTD

Each active route is represented by an XML element described by the following DTD:

<!ELEMENT USG (ROUTE)>
<!ATTLIST USG COMMAND CDATA #FIXED “ACTIVE_ROUTES”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT ROUTE (ID,
ADDRESS,MASK,GATEWAY,PORTNAME,INTERFACE,TYPE)>

<! -- ID indicates the ID of the route. The ID consists of two components separated by a slash (“/”). The first component, an unsigned interger, indicates the table id. The second component, also an integer, indicates the route id. -->
<!ELEMENT ID (#PCDATA)>

<!—ADDRESS contains the IP address of the destination network in dotted-decimal notation. -->
<!ELEMENT ADDRESS _STATE (#PCDATA)>

<!—MASK contains the address mask in dotted-decimal notation. -->
<!ELEMENT MASK (#PCDATA)>

<!—GATEWAY contains the IP address, in dotted-decimal notation, of the next hop to which packets should be sent en route to their final destination. -->
<!ELEMENT GATEWAY (#PCDATA)>

<!—PORTNAME contains the name of the port associated with the route (e.g. WAN, Eth1, Eth2, etc. -->
<!ELEMENT PORTNAME (#CDATA)>

<!—INTERFACE contains the name of the interface associated with the route (e.g. gei0, gei1, gei2, etc) -->
<!ELEMENT INTERFACE (#CDATA)>

<!—TYPE indicates the route type and is one of the following string values:

- “system” – The route was added automatically by the system.
- “static” – The route added was configured via a management interface (e.g. WMI) and added to the active routing table. The route is not retained in persistent storage and therefore not restored after a reboot.
- “persistent” - The route added was configured via a management interface (e.g. WMI) and added to the active routing table. The route is retained in persistent storage and is restored after a reboot. -->

<!ELEMENT TYPE (#PCDATA)>

Sample response for an active route list request :

```
<USG COMMAND="ACTIVE_ROUTES" VERSION="1.0">
        <ROUTE URI="api/routing/v1/activeRoutes/1/431">
                <ID>1/431</ID>
                <ADDRESS>0.0.0.0</ADDRESS>
                <MASK>0.0.0.0</MASK>
                <GATEWAY>67.130.148.254</GATEWAY>
                <PORTNAME>WAN</PORTNAME>
                <INTERFACE>gei0</INTERFACE>
                <TYPE>system</TYPE>
        </ROUTE>
</USG>
```

23. Persistent Routing Tables Commands

23.1 Listing Persistent Routes

Persistent routes can be listed via HTTP (s) “GET” requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/persistentRoutes[/table id][/route id]**. If the optional table id is specified, only those routes for the specified table are listed. If, in addition to a table id, a route id is specified, the single specified route is listed.

Each persistent route returned will be XML-encoded as defined by the Persistent Route DTD (see section 23.6)

23.2 Adding Persistent Routes

Persistent routes can be added via HTTP(s) “POST” requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/persistentRoutes/table id.** Note that a table id must be specified. The route ID will be assigned by the system.

The content of the request should be an XML-encoded representation of the route as defined by the Persistent Route DTD (see section 23.6)

Note however that ID and INTERFACE should be omitted from the XML-encoded representation of the route as these are assigned by the NSE.

23.3 Deleting Persistent Routes

Persistent routes can be deleted via HTTP(s) “DELETE” requests sent to the following Web address:

http[s]://NSE_URI/api/routing/v1/persistentRoutes/table id/route id. Note that only a single route can be deleted at a time. The route's table id and route id must be specified.

23.4 Table IDs

The system routing table has an ID of 254.

When the Load Balancing feature of the NSE is enabled, an additional routing table is created for each NSE Ethernet interface. These routing tables are used to route subscriber traffic. The routing table associated with the "WAN" labeled interface is used to route subscriber traffic for those subscribers assigned to WAN. Similarly, the routing table associated with the "Eth1" labeled interface is used to route subscriber traffic for those subscribers assigned to Eth1.

The IDs of these routing tables are: 1 for WAN, 2 for Eth1, 3 for Eth2, etc.

23.5 Route IDs

Each route has a unique id assigned by the system at the time of it's creation.

23.6 Persistent Route DTD

Each active route is represented by an XML element described by the following DTD:

<!ELEMENT USG (ROUTE)>
<!ATTLIST USG COMMAND CDATA #FIXED “PERSISTENT_ROUTES”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT ROUTE (ID,
ADDRESS,MASK,GATEWAY,PORTNAME,INTERFACE,ROLE)>

<! -- ID indicates the ID of the route. The ID consists of two components separated by a slash (“/”). The first component, an unsigned interger, indicates the table id. The second component, also an integer, indicates the route id. -->
<!ELEMENT ID (#PCDATA)>

<!—ADDRESS contains the IP address of the destination network in dotted-decimal notation. -->
<!ELEMENT ADDRESS _STATE (#PCDATA)>

<!—MASK contains the address mask in dotted-decimal notation. -->
<!ELEMENT MASK (#PCDATA)>

<!—GATEWAY contains the IP address, in dotted-decimal notation, of the next hop to which packets should be sent en route to their final destination. -->
<!ELEMENT GATEWAY (#PCDATA)>

<!—PORTNAME contains the name of the port associated with the route (e.g. WAN, Eth1, Eth2, etc. -->
<!ELEMENT PORTNAME (#CDATA)>

<!—INTERFACE contains the name of the interface associated with the route (e.g. gei0, gei1, gei2, etc) -->
<!ELEMENT INTERFACE (#CDATA)>

<!—ROLE indicates the route type and is one of the following string values:

- “wan” – The route will be active only when the associated interface is configured to be a WAN interface.
- “sub” – The route will be active only when the associated interface is configured to a Subscriber interface -->

<!ELEMENT ROLE (#PCDATA)>

Sample response for an persistent route list request :

```
<USG COMMAND="PERSISTENT_ROUTES" VERSION="1.0">
        <ROUTE URI="api/routing/v1/persistentRoutes/1/2">
                <ID>1/2</ID>
                <ADDRESS>1.1.3.0</ADDRESS>
                <MASK>255.255.255.0</MASK>
                <GATEWAY>67.130.148.1</GATEWAY>
                <PORTNAME>WAN</PORTNAME>
                <INTERFACE>gei0</INTERFACE>
                <ROLE>wan</ROLE>
        </ROUTE>
</USG>
```

24. Static Routing Tables Commands

24.1 Listing Static Routes

Static routes can be listed via HTTP (s) “GET” requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/staticRoutes[/table id][/route id]**. If the optional table id is specified, only those routes for the specified table are listed. If, in addition to a table id, a route id is specified, the single specified route is listed.

Each static route returned will be XML-encoded as defined by the Static Route DTD (see section 23.6) .

24.2 Adding Static Routes

Static routes can be added via HTTP(s) “POST” requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/staticRoutes/table id.** Note that a table id must be specified. The route ID will be assigned by the system.

The content of the request should be an XML-encoded representation of the route as defined by the Static Route DTD (see section 24.6)

Note however that ID and INTERFACE should be omitted from the XML-encoded representation of the route as these are assigned by the NSE.

24.3 Deleting Persistent Routes

Static routes can be deleted via HTTP(s) “DELETE” requests sent to the following Web address: **http[s]://NSE_URI/api/routing/v1/staticRoutes/table id/route id.** Note that only a single route can be deleted at a time. The route’s table id and route id must be specified.

24.4 Table IDs

The system routing table has an ID of 254.

When the Load Balancing feature of the NSE is enabled, an additional routing table is created for each NSE Ethernet interface. These routing tables are used to route subscriber traffic. The routing table associated with the "WAN" labeled interface is used to route subscriber traffic for those subscribers assigned to WAN. Similarly, the routing table associated with the "Eth1" labeled interface is used to route subscriber traffic for those subscribers assigned to Eth1.

The IDs of these routing tables are: 1 for WAN, 2 for Eth1, 3 for Eth2, etc.

24.5 Route IDs

Each route has a unique id assigned by the system at the time of it's creation.

24.6 Persistent Route DTD

Each active route is represented by an XML element described by the following DTD:

<!ELEMENT USG (ROUTE)>
<!ATTLIST USG COMMAND CDATA #FIXED “STATIC_ROUTES”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT ROUTE (ID,
ADDRESS,MASK,GATEWAY,PORTNAME,INTERFACE,ROLE)>

<! -- ID indicates the ID of the route. The ID consists of two components separated by a slash (“/”). The first component, an unsigned interger, indicates the table id. The second component, also an integer, indicates the route id. -->
<!ELEMENT ID (#PCDATA)>

<!—ADDRESS contains the IP address of the destination network in dotted-decimal notation. -->
<!ELEMENT ADDRESS _STATE (#PCDATA)>

<!—MASK contains the address mask in dotted-decimal notation. -->
<!ELEMENT MASK (#PCDATA)>

<!—GATEWAY contains the IP address, in dotted-decimal notation, of the next hop to which packets should be sent en route to their final destination. -->
<!ELEMENT GATEWAY (#PCDATA)>

<!—PORTNAME contains the name of the port associated with the route (e.g. WAN, Eth1, Eth2, etc. -->
<!ELEMENT PORTNAME (#CDATA)>

<!—INTERFACE contains the name of the interface associated with the route (e.g. gei0, gei1, gei2, etc) -->
<!ELEMENT INTERFACE (#CDATA)>

<!—ROLE indicates the route type and is one of the following string values:

- “wan” – The route will be active only when the associated interface is configured to be a WAN interface.
- “sub” – The route will be active only when the associated interface is configured to a Subscriber interface -->

<!ELEMENT ROLE (#PCDATA)>

Sample response for an static route list request :

```
<USG COMMAND="STATIC_ROUTES" VERSION="1.0">
        <ROUTE URI="api/routing/v1/persistentRoutes/1/2">
                <ID>1/2</ID>
                <ADDRESS>1.1.3.0</ADDRESS>
                <MASK>255.255.255.0</MASK>
                <GATEWAY>67.130.148.1</GATEWAY>
                <PORTNAME>WAN</PORTNAME>
                <INTERFACE>gei0</INTERFACE>
                <ROLE>wan</ROLE>
        </ROUTE>
</USG>
```

25. Class Based Queueing Commands

25.1 Listing Classes

Class Based Queueing classes can be listed via HTTP (s) "GET" requests sent to the following Web address:
http[s]://NSE_URI/api/cbQueueing/v1/class[/interface id][/class id]. If the optional interface id is specified, only those classes for the specified interface are listed. If, in addition to an interface id, a class id is specified, the single specified class is listed.

Each class returned will be XML-encoded as defined by the Class DTD (see section 23.6) .

25.2 Adding Classes

Classes can be added via HTTP(s) "POST" requests sent to the following Web address:
http[s]://NSE_URI/api/cbQueueing/v1/class/interface id. Note that an interface id must be specified. The class ID will be assigned by the system.

The content of the request should be an XML-encoded representation of the class as defined by the Class DTD (see section 24.6)

Note however that the ID should be omitted from the XML-encoded representation of the class as this is assigned by the NSE.

25.3 Deleting Classes

Classes can be deleted via HTTP(s) "DELETE" requests sent to the following Web address:
http[s]://NSE_URI/api/cbQueueing/v1/interface id/class id. Note that only a single class can be deleted at a time. The class's interface id and class id must be specified.

Note: only leaf classes can be deleted (i.e. classes with no child classes)

25.4 Interface IDs

Each Ethernet interface is assigned an ID. The "WAN" labeled interface is assigned the ID of "1". Its adjacent interface is assigned the ID of 2 and so on.

25.5 Class IDs

Each class has a unique id assigned by the system at the time of its creation.

25.6 Class DTD

Each Class is represented by an XML element described by the following DTD:

```
<!ELEMENT USG (CLASS)>
<!ATTLIST USG COMMAND CDATA #FIXED “CBQUEUEING_CLASS”>
<!ATTLIST USG VERSION CDATA “1.0”>

<!ELEMENT ROUTE (ID, NAME,  BANDWITDH_MIN_UP,  BANDWITDH_MAX_UP,
BANDWITDH_MIN_DOWN, BANDWITDH_MAX_DOWN,  RELATIVE_PRIORITY,
PARENT)>

<! -- ID indicates the ID of the class.  The ID consists of two components separated by a slash
(“/”).  The first component, an unsigned interger, indicates the interface id.  The second
component, also an integer, indicates the class id. -->
<!ELEMENT ID (#PCDATA)>

<!—NAME contains the name of the class. -->
<!ELEMENT NAME   (#CDATA)>

<!—BANDWITDH_MIN_UP contains the minimum bandwidth (in kbps) available to the class
in the upstream (to the network) direction. -->
<!ELEMENT BANDWITH_MIN_UP (#PCDATA)>

<!—BANDWITDH_MAX_UP contains the maximum bandwidth (in kbps) available to the class
in the upstream (to the network) direction -->
<!ELEMENT BANDWITH_MAX_UP (#PCDATA)>

<!—BANDWITDH_MIN_DOWN contains the minimum bandwidth (in kbps) available to the
class in the downstream (from the network) direction. -->
<!ELEMENT BANDWITH_MIN_DOWN (#PCDATA)>

<!—BANDWITDH_MAX_DOWN contains the maximum bandwidth (in kbps) available to the
class in the downstream (from the network) direction. -->
<!ELEMENT BANDWITH_MAX_DOWN(#PCDATA)>

<!—RELATIVE_PRIORITY Top level classes have 8 levels of priority (1 – 8).  All other classes
have 3 levels of priority (1-3).  A class’s absolute priority is a combination of its top level
ancestor’s priority and its own relative priority.  For example a class whose top level ancestor has
a priority of 4 and its own relative priority is 2 would have an absolute priority of 4.2  -->
<!ELEMENT RELATIVE_PRIORITY (#PCDATA)>

<!—PARENT specifies the classes parent class.  A class with no parent is a top level class.  The
class hierarchy (for each interface) is built using this field.   -->
<!ELEMENT PARENT (#CDATA)>
```

Sample response for a class list request :

```
<USG COMMAND="CBQUEUEING_CLASS" VERSION="1.0">
   <CLASS URI="api/cbQueueing/v1/class/1/65">
      <ID>1/65</ID>
      <NAME>priority1</NAME>
      <BANDWIDTH_MIN_UP>180000</BANDWIDTH_MIN_UP>
      <BANDWIDTH_MAX_UP>400000</BANDWIDTH_MAX_UP>
      <BANDWIDTH_MIN_DOWN>180000</BANDWIDTH_MIN_DOWN>
      <BANDWIDTH_MAX_DOWN>400000</BANDWIDTH_MAX_DOWN>
      <RELATIVE_PRIORITY>1</RELATIVE_PRIORITY>
   </CLASS>
</USG>
```

Contact Information:

Main:
Nomadix, Inc.
30851 Agoura Road,
Suite 102
Agoura Hills, CA 91301USA
+1.818.597.1500
http://www.nomadix.com/

Sales:
+1.818.597.1500
sales@nomadix.com

Technical Support:
+1.818.575.2590
support@nomadix.com