

IEEE P802.1Xck YANG Instance Document

IEEE 802 Plenary Meeting

Marc Holness
Version 5.0
13 July 2017

Introduction

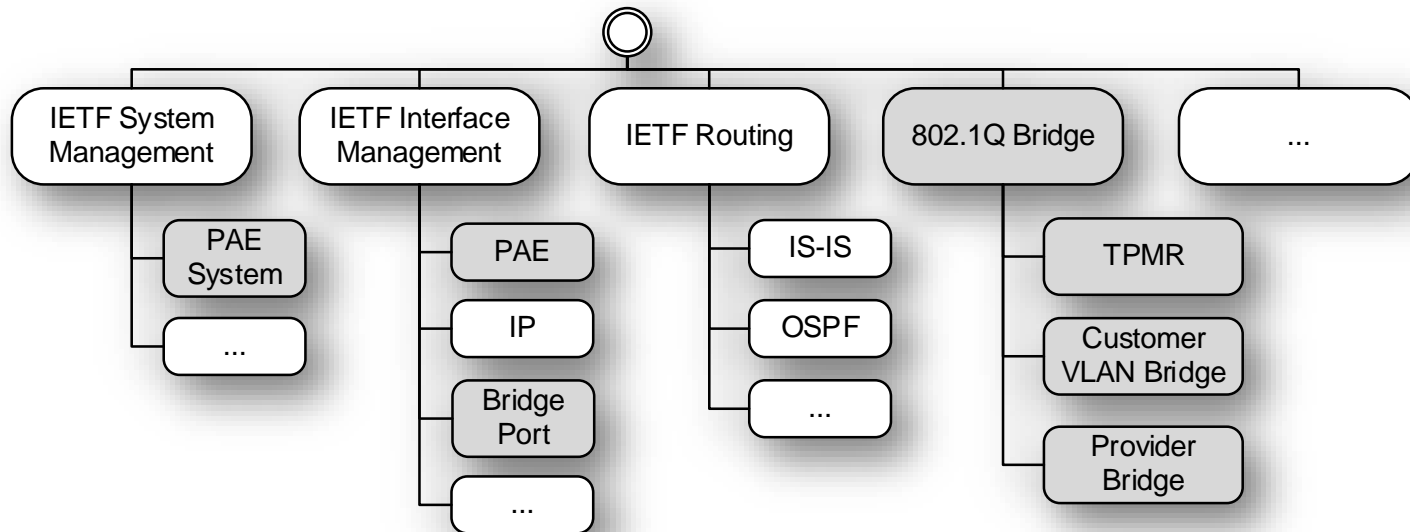


- This package provides “configuration instances” of various aspect of the P802.1Xck version D1.1 YANG modules

802.1 YANG Structure and Relationships

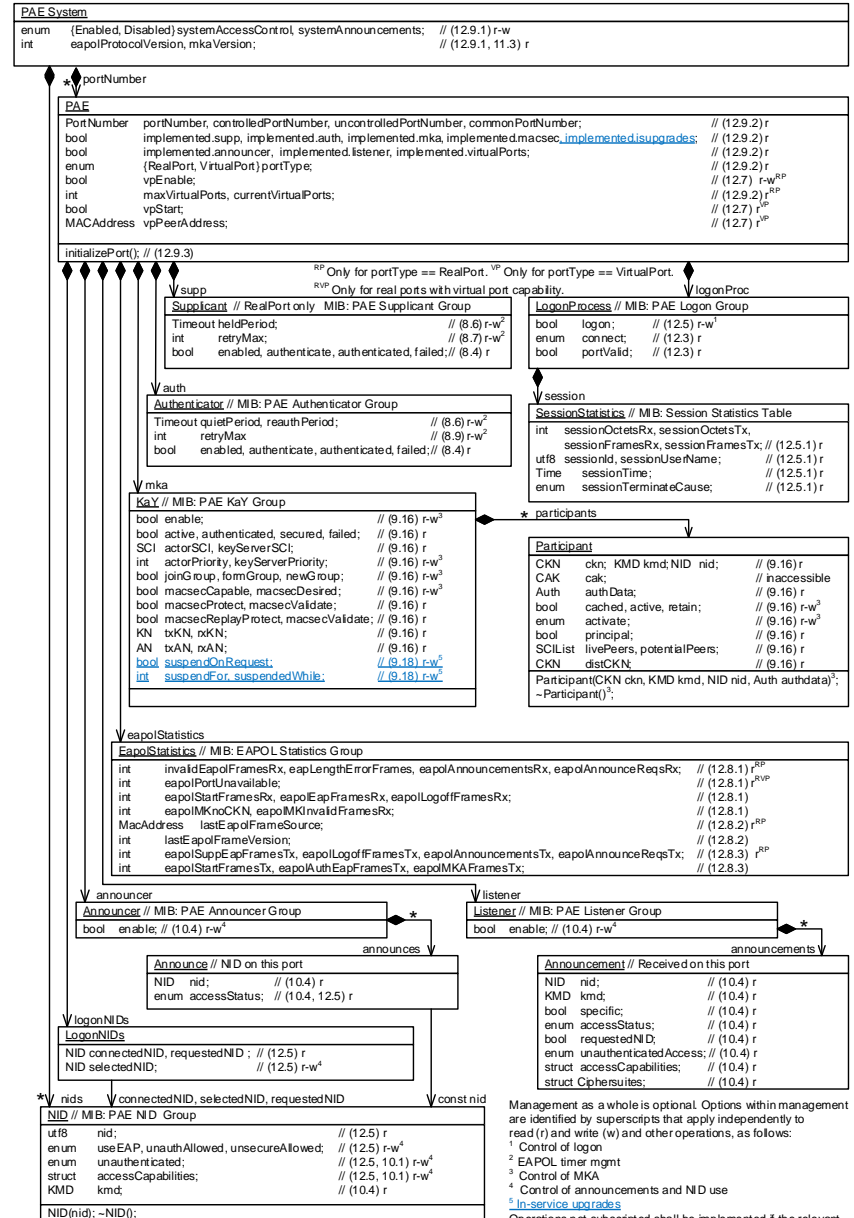


- The following hi-level YANG structure and relationships were defined



PAE Management Information Model

- The PAE management process controls and monitors the operation of each PAE's component protocol entities and processes, providing access to operational controls, statistics and diagnostic capabilities
- The management information that provides control over and reporting of these capabilities is summarized in the model to the right

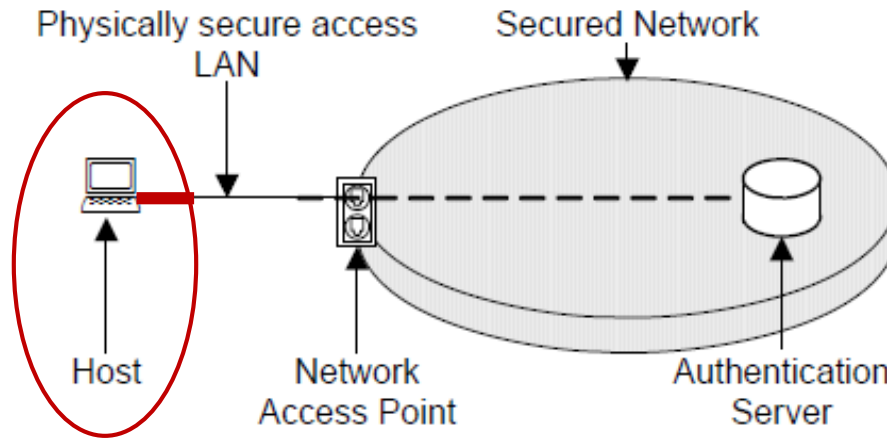


Management as a whole is optional. Options within management are identified by superscripts that apply independently to read (r) and write (w) and other operations, as follows:

- ¹ Control of logon
- ² EAPOL timer mgmt
- ³ Control of MKA
- ⁴ Control of announcements and NID use

^R In-service upgrades
 Operations not subscripted shall be implemented if the relevant capability is identified as implemented by the PAE.

Example 1: Host Accessing Network Access Point — Physically secure point-to-point LAN



Editor's Note: There is an `enable` attribute associated with the `supplicant`. This attribute is set by another process (e.g., EAPs). However, the YANG model will reflect the disposition of this state via the `enabled` attribute.

Editor's Note: Need to fix `supp` conditional statement in YANG module. For example,
`when "../port-capabilities/auth = 'true'" {`

NOTE: All NETCONF messages MUST be well-formed XML, encoded in UTF-8 [RFC3629].

Example 1: NETCONF Configuration



1

```
<rpc message-id="101">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <pae-system nc:operation="create">
        <name>aPAESystem</name>
        <system-access-control>enabled</system-access-control>
        <system-announcements>disabled</system-announcements>
      </pae-system>
      <interfaces>
        <interface>
          <name>if1</name>
          <type>ethernetCsmacd</type>
          <pae nc:operation="create">
            <pae-system>aPAESystem</pae-system>
            <port-type>real-port</port-type>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Provide `pae-system` data attributes, and
- Provide preliminary `pae` Interface data attributes
 - Pointer to `pae-system`
 - Set the `port-type`
- NOTE: Host 'logon process' is still *disabled* at this point

Example 1: NETCONF Configuration



2

```
<rpc message-id="102">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>

<rpc-reply message-id="102">
  <data>
    <top . . .>
      <paee>
        <port-capabilities>
          <supp>true</supp>
        </port-capabilities>
        <supplicant>
          <enabled>false</enabled>
        </supplicant>
      </paee>
    </top>
  </data>
</rpc-reply>
```

- Read the PAE `port-capabilities.supp` implemented indicator
 - A value of TRUE indicates a PACP EAP Supplicant is implemented
- Read the `supplicant enabled` data attributes
 - `Supplicant.enabled` is FALSE if the Port is not enabled, if the functionality provided by the PAE is not available, or not implemented, or the control variable enable has been cleared by management, e.g. because the application scenario authenticates a user and there is no user logged on

Example 1: NETCONF Configuration



3

```
<rpc message-id="103">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces>
        <interface>
          <name>if1</name>
          <pae>
            <logon-process>true</logon-process>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Enable the 'logon process'
 - For example, this would kick-start EAPs interactions

Example 1: NETCONF Configuration



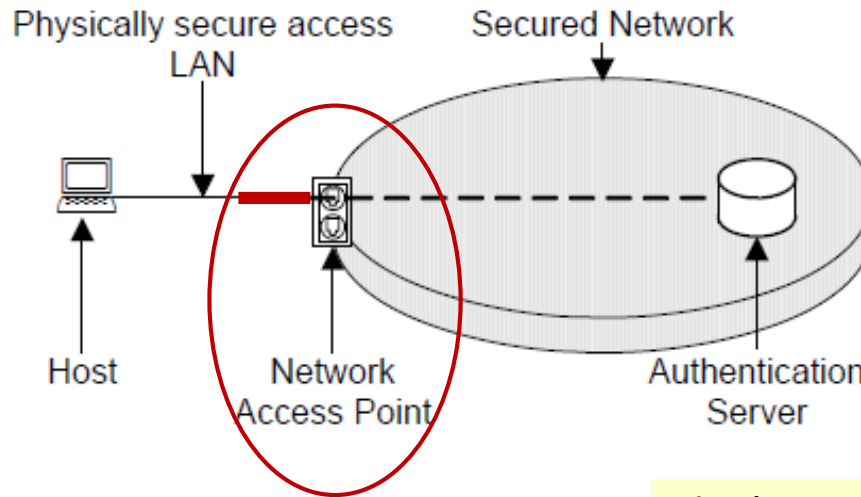
4

```
<rpc message-id="104">  
  <get-config>  
    <source>  
      <running/>  
    </source>  
    <filter . . ./>  
  </get-config>  
</rpc>
```

```
<rpc-reply message-id="104">  
  <data>  
    <top . . .>  
      <pae>  
        <supplicant>  
          <enabled>true</enabled>  
        </supplicant>  
      </pae>  
    </top>  
  </data>  
</rpc-reply>
```

- Read the *supplicant enabled* data attributes
 - *Supplicant.enabled* is now TRUE, which indicates that the 'logon process' has enabled the Supplicant PACP

Example 2: Network Access Point supporting Host — Physically secure point-to-point LAN



Editor's Note: There is an `enable` attribute associated with the `authenticator`. This attribute is set by another process (e.g., EAPs). However, the YANG model will reflect the disposition of this state via the `enabled` attribute.

NOTE: Need to fix `auth` conditional statement in the YANG module. For example,

```
when "../port-type = 'real-port' and
    ../port-capabilities/supp = 'true'"{
```

NOTE: All NETCONF messages MUST be well-formed XML, encoded in UTF-8 [RFC3629].

Example 2: NETCONF Configuration



1

```
<rpc message-id="101">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <pae-system nc:operation="create">
        <name>aPAESystem</name>
        <system-access-control>enabled</system-access-control>
        <system-announcements>disabled</system-announcements>
      </pae-system>
      <interfaces>
        <interface>
          <name>if1</name>
          <type>ethernetCsmacd</type>
          <pae nc:operation="create">
            <pae-system>aPAESystem</pae-system>
            <port-type>real-port</port-type>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Provide `pae-system` data attributes, and
- Provide preliminary `pae` Interface data attributes
 - Pointer to `pae-system`
 - Set the `port-type`
- NOTE: Network Access Point 'logon process' is still *disabled* at this point

Example 2: NETCONF Configuration



2

```
<rpc message-id="102">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>

<rpc-reply message-id="102">
  <data>
    <top . . .>
      <paes>
        <port-capabilities>
          <auth>true</auth>
        </port-capabilities>
        <authenticator>
          <enabled>false</enabled>
        </authenticator>
      </paes>
    </top>
  </data>
</rpc-reply>
```

- Read the PAE `port-capabilities.auth` implemented indicator
 - A value of TRUE indicates a PACP EAP Authenticator is implemented
- Read the `authenticator enabled` data attributes
 - `Authenticator.enabled` is FALSE if the Port is not enabled, if the functionality provided by the PAE is not available, or not implemented, or the control variable enable has been cleared by management, e.g. because the application scenario authenticates a user and there is no user logged on

Example 2: NETCONF Configuration



3

```
<rpc message-id="103">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces>
        <interface>
          <name>if1</name>
          <pae>
            <logon-process>true</logon-process>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Enable the 'logon process'
 - For example, this would kick-start EAPs interactions

Example 2: NETCONF Configuration



4

```
<rpc message-id="104">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>



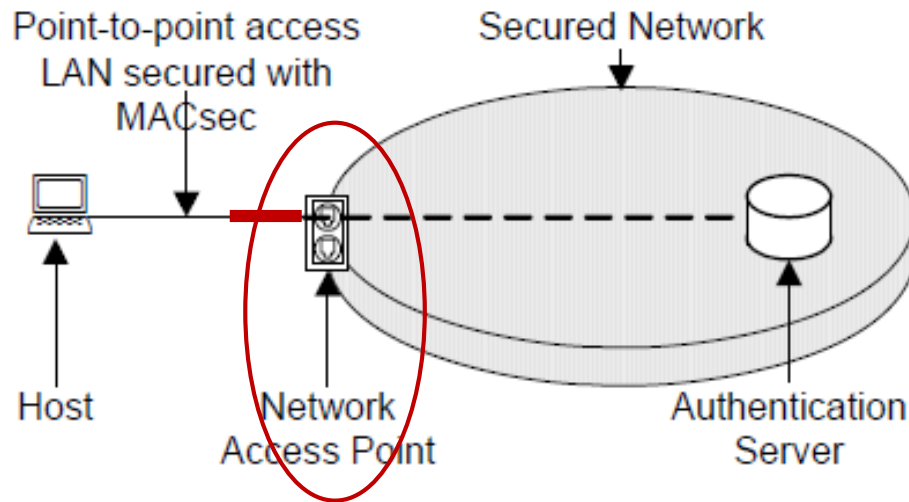
---



<rpc-reply message-id="104">
  <data>
    <top . . .>
      <paes>
        <authenticator>
          <enabled>true</enabled>
        </authenticator>
      </paes>
    </top>
  </data>
</rpc-reply>
```

- Read the `authenticator enabled` data attributes
 - `Authenticator.enabled` is now TRUE, which indicates that the 'logon process' has enabled the Authenticator PACP

Example 3: Network Access Point supporting Host — MACsec and point-to-point LAN



NOTE: All NETCONF messages MUST be well-formed XML, encoded in UTF-8 [RFC3629].

Example 3: NETCONF Configuration



1

```
<rpc message-id="101">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <pae-system nc:operation="create">
        <name>aPAESystem</name>
        <system-access-control>enabled</system-access-control>
        <system-announcements>disabled</system-announcements>
      </pae-system>
      <interfaces>
        <interface>
          <name>if1</name>
          <type>ethernetCsmacd</type>
          <pae nc:operation="create">
            <pae-system>aPAESystem</pae-system>
            <port-type>real-port</port-type>
            <kay>
              <enable>true</enable>
              <macsec>
                <capable>true</capable>
              </macsec>
            </kay>
            <listener>
              <enable>true</enable>
            </listener>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Provide `pae-system` data attributes, and
- Provide preliminary `pae` Interface data attributes
 - Pointer to `pae-system`
 - Set the `port-type` to `real-port`
 - Enable `kay`, `macsec`, and `listener`
- NOTE: Network Access Point 'logon process' is still *disabled* at this point

Example 3: NETCONF Configuration



2

```
<rpc message-id="102">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>
```

```
<rpc-reply message-id="102">
  <data>
    <top . . .>
      <paes>
        <port-capabilities>
          <auth>true</auth>
        </port-capabilities>
        <authenticator>
          <enabled>false</enabled>
        </authenticator>
      </paes>
    </top>
  </data>
</rpc-reply>
```

- Read the PAE `port-capabilities.auth` implemented indicator
 - A value of TRUE indicates a PACP EAP Authenticator is implemented
- Read the `authenticator enabled` data attributes
 - `Authenticator.enabled` is FALSE if the Port is not enabled, if the functionality provided by the PAE is not available, or not implemented, or the control variable enable has been cleared by management, e.g. because the application scenario authenticates a user and there is no user logged on

Example 3: NETCONF Configuration



3

```
<rpc message-id="103">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces>
        <interface>
          <name>if1</name>
          <pae>
            <logon-process>true</logon-process>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Enable the 'logon process'
 - For example, this would kick-start EAPs interactions

Example 3: NETCONF Configuration



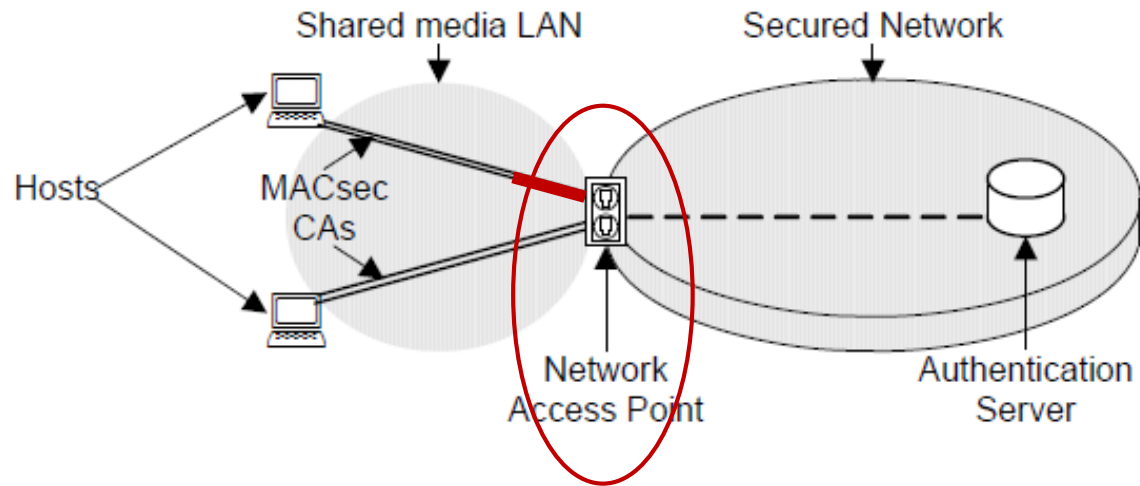
4

```
<rpc message-id="104">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>

<rpc-reply message-id="104">
  <data>
    <top . . .>
      <pae>
        <authenticator>
          <enabled>true</enabled>
        </authenticator>
      </pae>
    </top>
  </data>
</rpc>
```

- Read the `authenticator enabled` data attributes
 - `Authenticator.enabled` is now TRUE, which indicates that the 'logon process' has enabled the Authenticator PACP

Example 4: Network Access Point supporting Host — MACsec and multi-access LAN



NOTE: All NETCONF messages MUST be well-formed XML, encoded in UTF-8 [RFC3629].

Example 4: NETCONF Configuration



1

```
<rpc message-id="101">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <pae-system nc:operation="create">
        <name>aPAESystem</name>
        <system-access-control>enabled</system-access-control>
        <system-announcements>disabled</system-announcements>
      </pae-system>
      <interfaces>
        <interface>
          <name>if1</name>
          <type>ethernetCsmacd</type>
          <pae nc:operation="create">
            <pae-system>aPAESystem</pae-system>
            <port-type>virtual-port</port-type>
            <kay>
              <enable>true</enable>
              <macsec>
                <capable>true</capable>
              </macsec>
            </kay>
            <listener>
              <enable>true</enable>
            </listener>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Provide `pae-system` data attributes, and
- Provide preliminary `pae` Interface data attributes
 - Pointer to `pae-system`
 - Set the `port-type` to `virtual-port`
 - Enable `kay`, `macsec`, and `listener`
- NOTE: Network Access Point 'logon process' is still *disabled* at this point

Example 4: NETCONF Configuration



2

```
<rpc message-id="102">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>
```

```
<rpc-reply message-id="102">
  <data>
    <top . . .>
      <paes>
        <port-capabilities>
          <auth>true</auth>
        </port-capabilities>
        <authenticator>
          <enabled>false</enabled>
        </authenticator>
      </paes>
    </top>
  </data>
</rpc-reply>
```

- Read the PAE `port-capabilities.auth` implemented indicator
 - A value of TRUE indicates a PACP EAP Authenticator is implemented
- Read the `authenticator enabled` data attributes
 - `Authenticator.enabled` is FALSE if the Port is not enabled, if the functionality provided by the PAE is not available, or not implemented, or the control variable enable has been cleared by management, e.g. because the application scenario authenticates a user and there is no user logged on

Example 4: NETCONF Configuration



3

```
<rpc message-id="103">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces>
        <interface>
          <name>if1</name>
          <pae>
            <logon-process>true</logon-process>
          </pae>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

- Enable the 'logon process'
 - For example, this would kick-start EAPs interactions

Example 4: NETCONF Configuration



4

```
<rpc message-id="104">
  <get-config>
    <source>
      <running/>
    </source>
    <filter . . ./>
  </get-config>
</rpc>
```

```
<rpc-reply message-id="104">
  <data>
    <top . . .>
      <paе>
        <authenticator>
          <enabled>true</enabled>
        </authenticator>
      </paе>
    </top>
  </data>
</rpc-reply>
```

- Read the `authenticator enabled` data attributes
 - `Authenticator.enabled` is now TRUE, which indicates that the 'logon process' has enabled the Authenticator PACP