How to configure nonstandard AES 192/256 for a SNMPv3 user?

Avoid compatibility problems by choosing right authentication protocol

Different key extension algorithms cause compatibility issues with AES >128 protocols. To avoid them and to ensure best security, use appropriate authentication protocols to avoid key extension at all.

That leads to the simple rule:

For AES256 use at least SHA256!

Some devices* and SNMP tools use an AES key extension algorithm implementation for 192 and 256 bit key length that was not specified in the IETF draft http://tools.ietf.org/html/draft-blumenthal-aes-usm-04. Instead those implementations use the key extension algorithm specified by http://tools.ietf.org/html/draft-reeder-snmpv3-usm-3desede-00. To use the latter non-standard protocol follow the steps below:

1. Use SNMP4J 2.2.3 or later.
2. Add the nonstandard privacy protocol to the SecurityProtocols instance with

```
import org.snmp4j.security.nonstandard.PrivAES256With3DESKeyExtension;
SecurityProtocols.getInstance().addPrivacyProtocol(new  PrivAES256With3DESKeyExtension());
```

3. Specify the nonstandard privacy protocol for the SNMPv3 user that should use it:

```
    user = new UsmUser(new OctetString("SHAAES256"),
                               AuthSHA.ID,
                               new OctetString("SHAAES256AuthPassword"),
    // Use the following privacy protocol if you want to use AES 256 with 3DES  like key extension for
this user:
                               PrivAES256With3DESKeyExtension.ID,
    // Use the following privacy protocol for standard conform AES 256 privacy:
    //                         PrivAES256.ID,
                               new OctetString("SHAAES256PrivPassword"));
```

Note: Standard and non-standard protocols cannot be used for the same SNMPv3 security Name concurrently when using USM - you can use them in a command generator with DirectUserTarget.

** SNMP4J users reported that there are Cisco devices using the 3DES key extension also for AES.*