

## USING SHA-1 IN ORACLE 9i

Although the obfuscation toolkit of Oracle 9i does not support SHA (Secure Hashing Algorithm), we can instead use Java cryptography, which supports SHA-1. Here are the steps:

- 1) Confirm that Oracle's JVM is installed. Check to see if a key object already exists:

```
Desc sys.javasnm
```

- 2) Confirm these init.ora parameters:

```
SHARED_POOL_SIZE      >= 60Mb
JAVA_POOL_SIZE        >= 30Mb
The SYSTEM Tablespace >120Mb Of Free Space
The RBS Tablespace    > 250Mb Of Free Space
```

- 3) If the JVM is not already installed, run the **initjvm.sql** (run as sys.) This script is located in:

```
$ORACLE_HOME/javavm/install
```

- 4) Create the packages to be used:

```
Create Or Replace And Compile
  java source named "Shal" as
  import java.security.*;
  public class Shal {
    public static byte[] GetHash ( byte[] s ) throws
NoSuchAlgorithmException {
      MessageDigest sha = MessageDigest.getInstance("SHA-1");
      return sha.digest(s);
    }
  }
/

Create Or Replace Package shal Is
  Function gethash(p_str In Varchar2) Return Varchar2;
End shal;
/

Create Or Replace Package Body shal Is
  Function gethash_raw (p_raw In Raw) Return Raw
  As Language Java Name 'Shal.GetHash(byte[]) return byte[]';

  Function gethash(p_str In Varchar2) Return Varchar2 Is
  Begin
    Return RawToHex(gethash_raw(UtlRaw.Cast_To_Raw(p_str)));
  End gethash;
End shal;
/
```

- 5) The function is called "gethash." Here's an example of its usage:

```
Select shal.gethash('1234-5678-9012-3456') from dual;

SHA1.GETHASH('1234-5678-9012-3456')
-----
44AA647728E6FA5A0B7B098C1B50492B052BB91E
```