Technical Note:
RSA SecurID®/SA Integration

RSA SecurID® is a two-factor authentication system from RSA Security, Inc. (a division of EMC). Two-factor authentication is based on the concept of *something you know* (a password or PIN) and *something you have* (an authenticator) and provides stronger user authentication than passwords. This document describes how to take advantage of SecurID authentication in your SA system, it does not attempt to explain how to install, configure, or maintain RSA SecurID.

For detailed information about RSA SecurID, see *http://www.rsa.com*.

This document describes how SA authentication integrates with RSA SecurID. It assumes that you are already using RSA SecurID or will install it. An RSA SecurID server (RSA Authentication Manager or ACE Server) must be installed and fully configured before you can begin using SecurID authentication with SA.

RSA SecurID/SA Integration

SA users are required to authenticate to SA to perform any operations. SecurID integration allows them to use their existing RSA SecurID tokens for authentication. SA authentication can be seamlessly assimilated into your existing SecurID environment. As far as the RSA authentication server is concerned, the Web Services Data Access Engine (twist) server is just another SecurID agent.

SecurID support is automatic with the installation of an SA Core. Only a few configuration steps are required to take advantage of the feature:

The first two tasks must be performed on every Web Services Data Access Engine host in your Multimaster Mesh or in installations that have multiple installed Web Services Data Access Engines.

- Copying an RSA SecurID configuration file named `sdconf.rec` into a directory on any SA Core servers that host the Web Services Data Access Engine (twist). `sdconf.rec` is located on the RSA Authentication Manager/ACE Server host and contains required information about the RSA Authentication Manager that must be available to the SA Core.
- Shutting down the Web Services Data Access Engine(s) and restarting after editing the `loginModule.conf` file to enable SecurID authentication in SA.
- Creating/modifying users in the SAS Java Client or SAS Web Client to use SecurID authentication.

SA Support for SecurID Authentication Methods

RSA SecurID is based on two-factor authentication, with the SecurID token as the first factor and the Personal Identification Number (PIN) as the second factor.

The SecurID token is the *something you have* and the PIN is the *something you know*. These two factors offer much stronger authentication than a user password.

SecurID tokens can be either hardware-based (*hardware token* or *hard token*) or software-based (*software token* or *soft token*). The tokens provide a token code which, when combined with a pre-assigned (provisioned) PIN is known as a *passcode*.

Table 1-1 shows the most typical authentication methods and which are supported by SA/SecurID integration.

Table 1-1: SecurID Authentication Methods

AUTHENTICATION METHOD	DESCRIPTION
Normal Authentication	The most used method. The user's PIN is assigned (*provisioned*). The passcode is either accepted or rejected.
Next Tokencode Mode (Not supported)	This method is used when a user does not enter the passcode correctly. In Next Tokencode Mode, the user must wait for the tokencode to change and then submit the new tokencode. By default, a user will be put into the Next Tokencode Mode if the incorrect passcode for that user has been submitted three times consecutively.

Table 1-1: SecurID Authentication Methods (continued)

AUTHENTICATION METHOD	DESCRIPTION
New PIN Mode (Not supported)	This scenario occurs when the user must create a new PIN or modify an existing PIN.

Restrictions

RSA SecurID authentication is not an appropriate method for non-interactive scripts due to the fact the token code changes every 60 seconds and therefore will cause non-interactive scripts to fail. Your options are to rewrite the scripts to be interactive, or avoid using SecurID where such scripts would be affected.

SecurID/SA Integration Requirements

- Solaris
- Linux x86 and x86_64
- RSA ACE Server 6.1 or above.

Configuring SA/SecurID Integration

Support for RSA SecurID authentication is integrated into the SA Core and is installed when the SA Core is installed.

However, there are several configuration steps that you must complete to begin using RSA SecurID/SA authentication.

To use RSA SecurID authentication, the SA Core must know the IP address of the SecurID authentication server and be able to communicate with it in a secure manner.

If you have multiple slices installed in an SA core, the following steps must be performed for each Slice Component bundle host.

Phase 1: The RSA SecurID Authentication Configuration File

1. You must contact your RSA SecurID administrator and obtain the file:

   `sdconf.rec`

2. Copy this file to the following location on all servers in the core that host a Web Services Data Access Engine (twist):

   `/var/opt/opsware/crypto/twist`

3. Set the file permissions on each server to give the `twist` user ownership of this file and read privileges:

   ```
   chmod 400 /var/opt/opsware/crypto/twist/sdconf.rec
   chown twist /var/opt/opsware/crypto/twist/sdconf.rec
   ```

4. Ensure that there is no `securid` or `sdstatus.12` file in the `/var/opt/opsware/crypto/twist` directory. If either or both of these files exist, remove them.

Phase 2: Enable RSA SecurID Authentication in SA

1. By default, RSA SecurID authentication is not enabled. To enable it, on every server in the core that hosts a Web Services Data Access Engine (twist), shut down the component:

   ```
   /etc/init./opsware-sas stop twist
   ```

2. Locate the file:

   `/etc/opt/opsware/twist/loginModule.conf`

   Edit the file and add the line marked in bold in the example below:

   ```
   TruthLoginModule {
   com.opsware.login.SecurIDLoginModule sufficient debug=false
   next_tokencode_mode=false new_pin_mode=false;
   com.opsware.login.TruthLoginModule sufficient debug=false;
      };
   ```

3. Restart the Web Services Data Access Engine(s) on all servers:

   ```
   /etc/init./opsware-sas start twist
   ```

4. If you have multiple Slice Component bundles installed, stop the Command Center (OCC) server and HTTPs proxy on all other Slice Component bundle hosts.
5. At this point only the Command Center for the Slice Component bundle host that is being configured as the RSA server is running. Log into that host's OCC. This will generate the node secret (`securid` file) and the `sdstatus.12` file in the `/var/opt/opsware/crypto/twist` subdirectory as well as register the Slice Component bundle server with ACE.
6. You can now start the OCC and HTTPs proxies on all the other Slice Component bundle hosts in the Core.

Phase 3: Create/Modify SA Users to Use SecurID Authentication

Each user that is to use SecurID Authentication must first exist as an authenticated user in the RAS SecurID authentication server (ACE server) and then must either be created or modified in the SAS Client to use SecurID authentication.

In either the SAS Client or the SAS Web Client, on the user's Profile page, specify that the user's Credential Store should be **RSA 2-factor**.

For detailed information about creating or modifying users, see *Managing Users and User Groups* in the *SA Administration Guide*.

Troubleshooting

If you receive multiple `Authentication Failed` error messages, first check with your RSA SecurID administrator to insure that the user and passcode is still valid. If you are unable to solve the problem, contact your technical support representative.