Apple
Apple VPN Service

RSA SecurID Ready Implementation Guide

Last Modified: March 13, 2014

Partner Information

Product Information	
Partner Name	Apple
Web Site	www.apple.com
Product Name	Apple VPN Service
Version & Platform	Mac OS X Server 10.8.x
Product Description	Mac OS X Server provides a Virtual Private Network (VPN) service allowing users to access their corporate network over the Internet. The VPN service currently supports L2TP Over IPSec and PPTP protocols.

Solution Summary

Virtual Private Network (VPN) access enables users to take advantage of network services while they're offsite and simultaneously prevents access by unauthorized individuals. Mac OS X Server supports standards-based L2TP/IPSec and PPTP tunneling protocols to provide encrypted VPN connections for Mac OS, Windows, and iOS devices. These VPN services use secure authentication methods, including RSA SecurID authentication.

Important: RSA SecurID authentication is not supported on Mac OS X Server 10.6.x "Snow Leopard". See the Known Issues sections for more information.

At the time of this writing, RSA SecurID authentication is only supported on the Apple VPN Service when running on the following operating systems:

Mac OS X Server 10.5.x "Leopard"

Mac OS X Server 10.7.x "Lion"

Mac OS X Server 10.8.x "Mountain Lion"

RSA Authentication Manager supported features Apple VPN Service on Mac OS X Server 10.8.3	
RSA SecurID Authentication via Native RSA SecurID Protocol	Yes
RSA SecurID Authentication via RADIUS Protocol	No
On-Demand Authentication via Native SecurID Protocol	Yes
On-Demand Authentication via RADIUS Protocol	No
Risk-Based Authentication	No
Risk-Based Authentication with Single Sign-On	No
RSA Authentication Manager Replica Support	Yes
Secondary RADIUS Server Support	No
RSA SecurID Software Token Automation	No
RSA SecurID SD800 Token Automation	No
RSA SecurID Protection of Administrative Interface	No

Authentication Agent Configuration

Authentication Agents are records in the RSA Authentication Manager database that contain information about the systems for which RSA SecurID authentication is provided. All RSA SecurID-enabled systems require corresponding Authentication Agents. Authentication Agents are managed using the RSA Security Console.

The following information is required to create an Authentication Agent:

- Hostname
- IP Addresses for network interfaces

Set the Agent Type to "Standard Agent" when adding the Authentication Agent. This setting is used by the RSA Authentication Manager to determine how communication with the Apple VPN Service will occur.

RSA SecurID files

RSA SecurID Authentication Files	
Files	**Location**
sdconf.rec	/var/ace
Node Secret	/var/ace
sdstatus.12	/var/ace
sdopts.rec	/var/ace

Note: The appendix of this document contains more detailed information regarding these files.

Partner Product Configuration

Before You Begin

This section provides instructions for configuring the Apple VPN Service with RSA SecurID Authentication. This document is not intended to suggest optimum installations or configurations.

It is assumed that the reader has both working knowledge of all products involved, and the ability to perform the tasks outlined in this section. Administrators should have access to the product documentation for all products in order to install the required components.

All Apple VPN Service components must be installed and working prior to the integration. Perform the necessary tests to confirm that this is true before proceeding.

Enabling RSA SecurID Authentication on the Apple VPN Service

The Apple VPN Service is part of a Mac OS X Server installation. The Server application (or Server Admin on Mac OS 10.5.x) may be used to configure standard VPN services, but this application does not have an interface for choosing the RSA SecurID Authentication method. Instead, RSA SecurID authentication must be configured via the command line using the **serveradmin** command line utility.

1. Create a directory called **/var/ace** on the VPN Server.

```
# sudo mkdir /var/ace
```

2. Copy the **sdconf.rec** file obtained from your RSA Authentication Manager deployment into this directory. You may be prompted to authenticate as an administrator to allow the copy.
3. Enable the RSA EAP-SecurID authentication method for the desired VPN protocols.

For PPTP:

```
# sudo serveradmin settings
vpn:Servers:com.apple.ppp.pptp:PPP:AuthenticatorEAPPlugins:_array_index: 0 =
"EAP-RSA"

# sudo serveradmin settings
vpn:Servers:com.apple.ppp.pptp:PPP:AuthenticatorProtocol:_array_index:0 = "EAP"
```

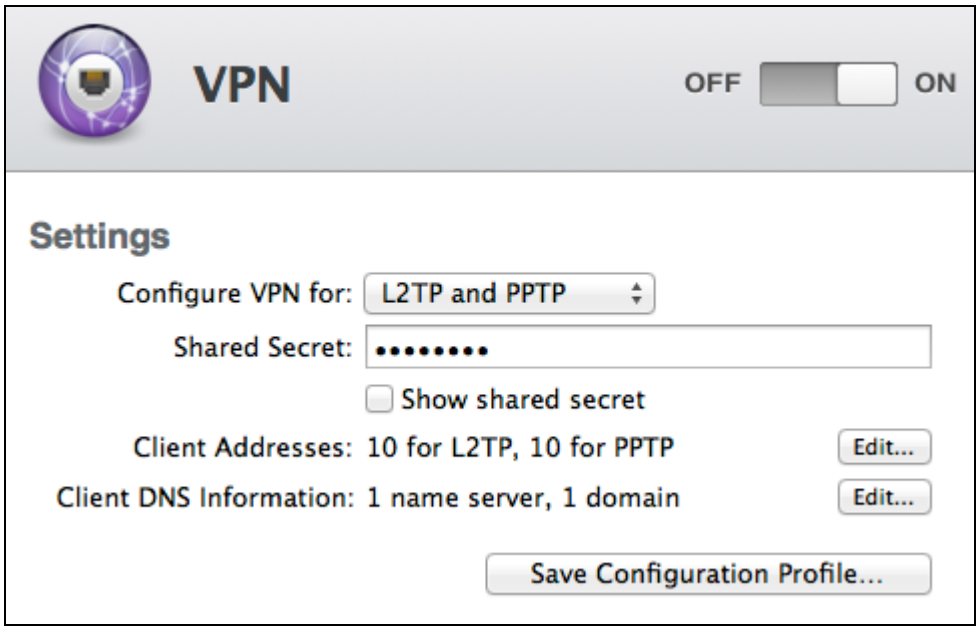
For L2TP over IPSec:

```
# sudo serveradmin settings
vpn:Servers:com.apple.ppp.l2tp:PPP:AuthenticatorEAPPlugins:_array_index: 0 =
"EAP-RSA"

# sudo serveradmin settings
vpn:Servers:com.apple.ppp.l2tp:PPP:AuthenticatorProtocol:_array_index:0 = "EAP"
```

4. Using the **Server** application, locate the the **Accounts > Users** section. Make sure that users are configured for each of your SecurID users. The username should match what is configured on the RSA Authentication Manager.

5. In the **Services > VPN** section, ensure the VPN Service is configured for the VPN protocols that are now configured to use RSA SecurID authentication. If **L2TP** is used, specify a **Shared Secret** to use for machine authentication. If the VPN Service was running when you configured RSA SecurID authentication, restart the service.



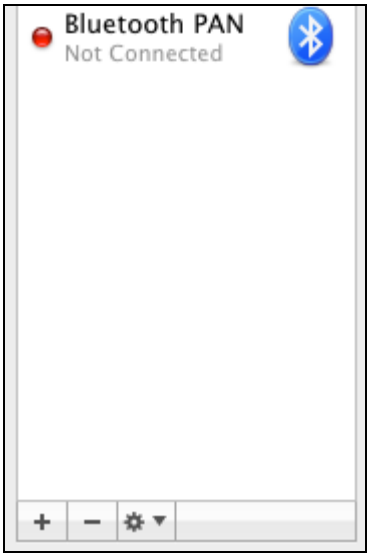
Note: The user interface for Mac OS X Leopard Server's Server Admin application is different from the screenshots presented in this guide, but the configuration steps are the same.

If you're unsure how to configure the VPN service using the Server Admin application, refer to the documentation for Mac OS X Leopard Server.

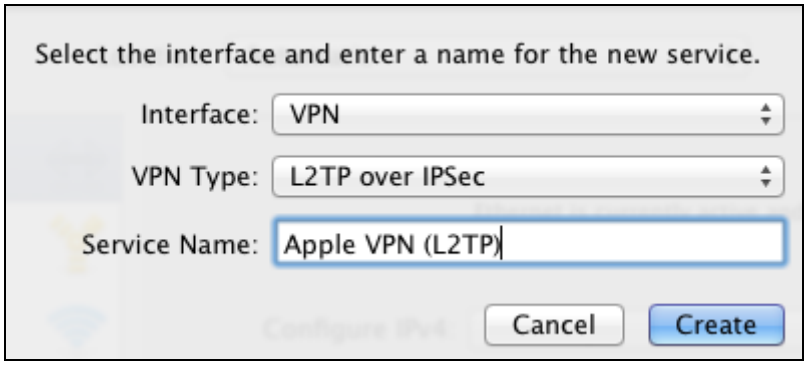
Enabling RSA SecurID in the Mac OS X VPN Client

The Mac OS X VPN Client is installed by default during a normal installation of Mac OS X. Follow the instructions below to enable the Apple VPN Client to connect using RSA SecurID authentication.

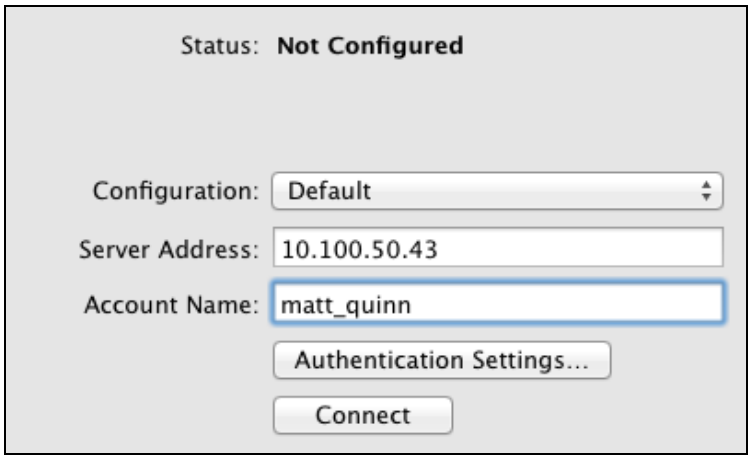
1. Begin by opening the **Network Preferences** (**System Preferences > Network**).
2. Click the **plus symbol** (**+**) to add a new interface.



3. Select **VPN** from the **Interface** drop down menu. Select the **VPN Type** (either **L2TP over IPSec** or **PPTP**) to match what is configured on the Apple VPN Server. Give the Interface a name and click **Create**.



4. Back on the Network Preferences screen, configure the newly created VPN Connection. Provide the IP address of the VPN Server in the **Server Address** field. For **Account Name**, provide the SecurID username of the account that will be used to authenticate. Click **Authentication Settings**.



5. Select **RSA SecurID** as the **User Authentication** type. If using **L2TP over IPSec**, provide the **Shared Secret** configured at the VPN Server. Click **OK** when finished.

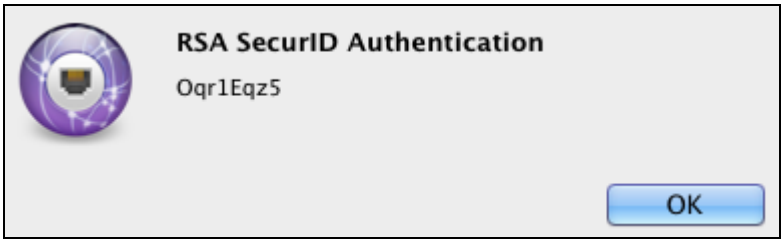
6. Click **Apply** to save the configuration changes. This connection can now be used to establish a VPN connection to the Apple VPN Server. When connecting, the user will be prompted for their SecurID username and passcode to prove their identity to the server.

RSA SecurID Login Screens

Login screen:

User-defined New PIN:

System-generated New PIN:



Next Tokencode:

Certification Checklist for RSA Authentication Manager

Date Tested: March 13, 2014

Certification Environment		
Product Name	**Version Information**	**Operating System**
RSA Authentication Manager	8.1	Virtual Appliance
Apple VPN Service	10.8 Build 12A269	Mac OS X Server 10.8.3
Apple iPad 2		iOS 6.1.3

Mandatory Functionality			
RSA Native Protocol		**RADIUS Protocol**	
New PIN Mode			
Force Authentication After New PIN	✓	Force Authentication After New PIN	N/A
System Generated PIN	✓	System Generated PIN	N/A
User Defined (4-8 Alphanumeric)	✓	User Defined (4-8 Alphanumeric)	N/A
User Defined (5-7 Numeric)	✓	User Defined (5-7 Numeric)	N/A
Deny 4 and 8 Digit PIN	✓	Deny 4 and 8 Digit PIN	N/A
Deny Alphanumeric PIN	✓	Deny Alphanumeric PIN	N/A
Deny PIN Reuse	✓	Deny PIN Reuse	N/A
Passcode			
16-Digit Passcode	✓	16-Digit Passcode	N/A
4-Digit Fixed Passcode	✓	4-Digit Fixed Passcode	N/A
Next Tokencode Mode			
Next Tokencode Mode	✓	Next Tokencode Mode	N/A
On-Demand Authentication			
On-Demand Authentication	✓	On-Demand Authentication	N/A
On-Demand New PIN	✓	On-Demand New PIN	N/A
Load Balancing / Reliability Testing			
Failover (3-10 Replicas)	✓	Failover	N/A
No RSA Authentication Manager	✓	No RSA Authentication Manager	N/A

GLS

✓ = Pass ✗ = Fail N/A = Not Applicable to Integration

Known Issues

RSA SecurID Authentication does not work with Mac OS X 10.6.x "Snow Leopard" Server.

Due to a bug in Apple's VPN implementation, RSA SecurID Authentication does not work properly with the VPN server built in to Mac OS X 10.6.x. The VPN server sends incorrectly formed authentication packets which the RSA Authentication Manager is unable to decrypt, resulting in authentication failures.

Apple is aware of this issue, and has corrected it in Mac OS X Server 10.7.x, but has not released a fix for Mac OS X Server 10.6.x.

There is no workaround for this issue. The only solution is to use a compatible version of Mac OS X:

Mac OS X 10.5.x "Leopard"

Mac OS X 10.7.x "Lion"

Mac OS X 10.8.x "Mountain Lion"

Appendix

Partner Integration Details	
RSA SecurID API	5.0.3.2
RSA Authentication Agent Type	Standard Agent
RSA SecurID User Specification	Designated Users
Display RSA Server Info	No
Perform Test Authentication	No
Agent Tracing	No

Node Secret:

The node secret is stored in a file called **securid** in the **/var/ace** directory. To clear the node secret on the VPN Server, delete this file.

sdconf.rec:

The sdconf.rec configuration file is stored in the **/var/ace** directory. To refresh this file, replace it with a current copy obtained from RSA Authentication Manager.

sdopts.rec:

The optional sdopts.rec configuration file can be staged in the **/var/ace** directory. This file contains optional parameters that modify the behavior of the RSA Authentication Agent library. Refer to the **RSA Authentication Agent API Developer's Guide** for more information on these parameters and when they should be used.

sdstatus.12:

This file is generated automatically by the Authentication Agent library and is found in **/var/ace**.