National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme
Validation Report

iAnywhere™ Solutions, Inc – A Sybase Company
Dublin, CA

Adaptive Server Anywhere 9.0.1/9.0.2 Component
of SQL Anywhere Studio 9

Report Number:	**CCEVS-VR-06-0018**
Dated:	**24 April 2006**
Version:	**1.0**

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6740
Fort George G. Meade, MD 20755-6740

ACKNOWLEDGEMENTS

Validation Team

Mr. Daniel P. Faigin
The Aerospace Corporation
El Segundo, California

The Validation Team also thanks Ms. Nicole Carlson, *The Aerospace* Corporation, for the work she performed when she was Lead Validator, and Mr. Kenneth Elliott for his work as Senior Validator.

Common Criteria Testing Laboratory

Ms. Tammy Compton, Lead Evaluator
Mr. Mike Boberski
Science Applications International Corporation
Columbia, Maryland

Much of the material in this report was extracted from evaluation material prepared by the CCTL. The CCTL team deserves credit for their hard work in developing that material. Many of the product descriptions in this report were extracted from the Adaptive Server Anywhere Security Target.

Table of Contents

1 Executive Summary

This report documents the assessment of the National Information Assurance Partnership (NIAP) validation team of the evaluation of the Adaptive Server Anywhere 9.0.1/9.0.2 component of the Sybase SQL Anywhere Studio 9 product.[1] It presents the evaluation results, their justifications, and the conformance results. This Validation Report is not an endorsement of the Target of Evaluation by any agency of the U.S. government, and no warranty is either expressed or implied.

The evaluation was performed by the Science Applications International Corporation (SAIC) Common Criteria Testing Laboratory (CCTL) in Columbia, Maryland, United States of America, and was completed in April 2006. The information in this report is largely derived from the Evaluation Technical Report (ETR) and associated test reports, all written by SAIC. The evaluation determined that the product is both **Common Criteria Part 2 Extended and Part 3 Conformant**, and meets the assurance requirements of Evaluation Assurance Level (EAL) 3 augmented with ALC_FLR.2.

Adaptive Server Anywhere (ASA) is a relational database management system (RDBMS). According to the vendor, it was designed to support multiple operating systems as well as operate efficiently with limited memory, CPU power, and disk space. Non-security relevant capabilities of the product include full transaction processing, referential integrity, SQL stored procedures,[2] triggers, row-level locking, automatic event scheduling and automatic recovery. Core features such as the query optimizer and the data caching mechanism are designed specifically to operate with minimal resources. At the same time, ASA contains the features needed to take advantage of workgroup servers, including support for many users, scalability over multiple CPUs, and advanced concurrency features.[3] ASA is designed to be self-tuning and yet maintain a small footprint. ASA symmetric multi-processor (SMP) support ensures top performance for greater numbers of users. A high-performance, self-tuning query optimizer determines the most effective way to access information and utilize additional processors, thereby improving performance and eliminating the need for expert tuning.

The Adaptive Server Anywhere RDBMS was originally created by Watcom and was called Watcom SQL. Watcom was acquired by PowerSoft in 1993, and the product was included with Powersoft’s visual programming environment PowerBuilder. PowerSoft released Watcom SQL version 4.0. In 1995, PowerSoft and Sybase merged and soon a new version was released under the name Sybase SQL Anywhere 5.0. Version 6.0 was renamed Adaptive Server Anywhere. Version 7.0.0 of Adaptive Server Anywhere received a C2 rating under the Trusted Technologies Assessment Program (TTAP-CSC-EPL-00-002) in September 2000. Version 9.0.2 is the latest version.

[1] Henceforth refered to as “ASA”. This product must be configured in accordance with the Common Criteria configuration instructions, as detailed in Table 2-1 (Page 3).

[2] The product also supports Java stored procedures, but the support for Java is not installed in the evaluated configuration.

[3] Note that the claims of non-security features are derived from product documentation, and have not been verified by the evaluation. They are included here solely for descriptive purposes.

Adaptive Server Anywhere is part of the Sybase SQL Anywhere Studio 9 product. This product is a collection of tools and databases designed for server, desktop, mobile, and remote office environments. The product includes not only the Adaptive Server Anywhere database, but the UltraLite database, MobiLink synchronization, QAnywhere, SQL Remote, and design and administration tools and utilities. **It is important to note that the only component of SQL Anywhere Studio that was evaluated was Adaptive Server Anywhere**. Any other components included with the product package, and any security claims made for those products in advertising material, *have not been verified* as part of this evaluation. Further, in the evaluated configuration, support for Java is *not* installed, and claims regarding Java have *not* been evaluated.

The focus of the evaluation is the ASA Server, which is accessed through two application-level protocols, the Command Sequence protocol (CmdSeq) and the Tabular Data Stream (TDS) Protocol. The protocols are used by untrusted client processes, via routines in provided libraries, to communicate with the Server. Administrators interface with the ASA Server via utility programs provided to facilitate ASA administration. These utility programs use available library routines, just like other untrusted clients, to invoke the protocols and communicate with the ASA Server.

The Adaptive Server Anywhere RDBMS is available for a wide variety of Windows and Unix platforms. *Only a subset of these platforms have been tested and meet the stated requirements for the IT Environment.*

This validation assumes the TOE has been configured as described in *Supplement for Installing Adaptive Server Anywhere for Common Criteria Configuration* [8]. Note that this configuration guide must be downloaded and followed to ensure the product is in the evaluated configuration; information on the Common Criteria configuration *is not* included in the packaged product itself.

The Target of Evaluation (TOE) identified in this Validation Report has been evaluated at a NIAP approved Common Criteria Testing Laboratory using the Common Methodology for IT Security Evaluation (Version 1.0) for conformance to the Common Criteria for IT Security Evaluation (Version 2.1). This Validation Report applies only to the specific version of the TOE as evaluated. The evaluation has been conducted in accordance with the provisions of the NIAP Common Criteria Evaluation and Validation Scheme and the conclusions of the testing laboratory in the evaluation technical report are consistent with the evidence provided.

The validation team monitored the activities of the evaluation team, observed evaluation testing activities, provided guidance on technical issues and evaluation processes, and reviewed the individual work units and successive versions of the ETR. The validation team found that the evaluation showed that the product satisfies all of the functional requirements and assurance requirements stated in the Security Target (ST). Therefore the validation team concludes that the testing laboratory's findings are accurate, the conclusions justified, and the conformance results are correct. The conclusions of the testing laboratory in the evaluation technical report are consistent with the evidence produced.

The SAIC evaluation team concluded that the Common Criteria requirements for Evaluation Assurance Level 3, augmented with ALC_FLR.2, have been met.

The technical information included in this report was obtained from the Evaluation Technical Report (ETR) Part 1 (non-proprietary) produced by SAIC, the Adaptive Server Anywhere Security Target, and research and analysis performed by the Validation Team.

2 Identification

The CCEVS is a joint National Security Agency (NSA) and National Institute of Standards effort to establish commercial facilities to perform trusted product evaluations. Under this program, security evaluations are conducted by commercial testing laboratories called Common Criteria Testing Laboratories (CCTLs) using the Common Evaluation Methodology (CEM) for Evaluation Assurance Level (EAL) 1 through 4 in accordance with National Voluntary Laboratory Assessment Program (NVLAP) accreditation.

The NIAP Validation Body assigns Validators to monitor the CCTLs to ensure quality and consistency across evaluations. Developers of information technology products desiring a security evaluation contract with a CCTL and pay a fee for their product's evaluation. Upon successful completion of the evaluation, the product is added to NIAP's Validated Products List.

Table 1 provides information needed to completely identify the product, including:

- The Target of Evaluation (TOE): the fully qualified identifier of the product as evaluated.
- The Security Target (ST), describing the security features, claims, and assurances of the product.
- The conformance result of the evaluation.
- The Protection Profile to which the product is conformant.
- The organizations and individuals participating in the evaluation.

Table 2-1. Evaluation Identifiers

Item	Identifier
Evaluation Scheme	United States NIAP Common Criteria Evaluation and Validation Scheme
TOE:	The Adaptive Server Anywhere (versions 9.0.1 or 9.0.2) component of the Sybase SQL Anywhere Studio 9 product , configured in accordance with *Supplement for Installing Adaptive Server Anywhere for Common Criteria Configuration* [8]. Specifically, the evaluated versions of the product are: • Adaptive Server Anywhere version 9.0.2 build 3221 for Microsoft Windows XP, Windows 2000, and Windows 2003 Server. • Adaptive Server Anywhere version 9.0.2 build 3219 for Sun Solaris 8, and Redhat Linux Advanced Server 2.1.

Item	Identifier
	• Adaptive Server Anywhere version 9.0.1 build 2085 for Microsoft Windows XP, Windows 2000, Windows 2003 Server, Sun Solaris 8, and Redhat Linux Advanced Server 2.1. The Adaptive Server Anywhere TOE also consists of the following Guidance Documents: • *Supplement for Installing Adaptive Server Anywhere for Common Criteria Configuration*, Document ID: DC00080-01-1252-01, Last revised: April 2006. Available at http://www.ianywhere.com/developer/ product_manuals/sqlanywhere/sqlanywhere_cc_configuration.pdf. • *Sybase ASA SQL reference:* DC38129-01-0901-01, version 9.0.1, January 2004 DC38129-01-0902-01, version 9.0.2, October 2004 • *Sybase ASA Error Messages:* DC38131-01-0901-01, version 9.0.1, January 2004 DC38131-01-0902-01, version 9.0.2, October 2004 • *Sybase ASA Database Administration Guide:* DC38123-01-0901-01, version 9.0.1, January 2004 DC38123-01-0902-01, version 9.0.2, October 2004 • *SQL Anywhere Studio Security Guide:* DC38177-01-0901-01, version 9.0.1, January 2004 DC38177-01-0902-01, version 9.0.2, October 2004 • *Sybase Adaptive Server Anywhere Delivery and Operation Procedures*, Revision 0.1, May 26, 2004
Protection Profile	The ST contains no claim of PP compliance
ST:	*Adaptive Server Anywhere Security Target,* Version 1.0, April 11, 2006
Evaluation Technical Report	• *Evaluation Technical Report for the Sybase Adaptive Server Anywhere, Versions 9.0.1 and 9.0.2, Part I (Non-Proprietary)*, Version 7.0, April 18, 2006 • *Evaluation Technical Report for the Sybase Adaptive Server Anywhere, Versions 9.0.1 and 9.0.2, Part II (Proprietary)*, Version 5.0, April 18, 2006 • *Evaluation Team Test Plan for the Sybase Adaptive Server Anywhere, ETR Part II Supplement (SAIC and Sybase Proprietary),* Version 2.0, 17 February 2006.
CC Version	Common Criteria for Information Technology Security Evaluation, Version 2.1
Conformance Result	CC Part 2 extended, CC Part 3 conformant
Sponsor	iAnywhere Solutions, Inc, A Sybase Company, Dublin, CA, USA
Developer	iAnywhere Solutions, Inc, A Sybase Company, Dublin, CA, USA
Common Criteria Testing Lab (CCTL)	SAIC, Columbia, MD, USA
CCEVS Validators	Daniel P. Faigin, The Aerospace Corporation, El Segundo, CA

3 Security Policy

The Security Functional Policies (SFPs) implemented by Adaptive Server Anywhere permit protection of user data, provide for authenticated user access, provide accountability for actions, and protect the mechanism that provides the security policies.

Note: Much of the description of the Adaptive Server Anywhere security policy has been extracted and reworked from the Adaptive Server Anywhere Security Target.

3.1 User data protection

3.1.1 Access Control Policies

The ASA Server implements a discretionary access control (DAC) policy that covers the following objects: tables, views, stored procedures and user-defined functions. This policy controls access to these objects by any database subject based on the user identity, group membership(s), and authorities of the subject, and the ownership and Access Control List (ACL) of the object. The ASA server supports other objects, but these objects are either (a) PUBLIC objects (e.g., global variables and messages), private objects (e.g., temporary tables, connection variables), or derive their protection from one of the protected objects from which they are associated (e.g., triggers and defaults).

Access permissions for the controlled objects are stored in ACLs associated with each objects, and are maintained internally in system database tables. Each ACL entry either GRANTs or REVOKEs permissions for a user or group; the set of permissions that can be granted or revoked are as follows:

1. For tables and views: INSERT, SELECT, UPDATE, DELETE, ALTER (tables only), and REFERENCES (tables only)
2. For stored procedures and user-defined functions: EXECUTE.

The ASA server uses these permissions, in conjunction with the object's ownership and subject's characteristics to determine access. The algorithm used is as follows:

1. If the user ID has DBA authority, the user ID can carry out any action in the database.
2. Otherwise, permission depends on the permissions assigned to the individual user. If the user ID has been granted permission to carry out the action, then the action proceeds.
3. If no individual settings have been made for that user, permission depends on the permissions of each of the groups to which the member belongs. If any of these groups has permission to carry out the action, the user ID has permission by virtue of membership in that group, and the action proceeds.

Only an authorized administrator can create or delete a database. The authorized administrator can subsequently GRANT users the permission to create tables, views, and stored procedures, as well as other capabilities in the database. When an authorized administrator or user (with RESOURCE authority) creates a table, the user becomes the table owner and inherits the ability to perform any operation on that table.

If a user is granted a particular access to a view, this implies that the user must also have the same permission on all objects upon which the view depends (via the view only), provided the owner of the view also has the appropriate permission on all of the objects upon which the view depends.

For stored procedures and user-defined functions the only permission is execute. If a user has execute permission on a stored procedure, the user can access all objects referenced by the stored procedure, provided the owner of the stored procedure or user-defined function has the appropriate permission on them.

Permissions can be granted with or without the GRANT option. The GRANT option controls the ability to propagate the permission; if present, it permits the subject to subsequently grant the specific permission to other users.

3.1.2 Users and Groups

While user identities can be used in ACLs to assign specific access permissions to specific users, ASA also supports a “group” feature. A group is a special user identity that is allowed to have members. Both users and groups can be members of groups, and each user or group can be a member of multiple groups. Membership in a group can be granted by the authorized administrator or by the group’s user ID. Groups provide a convenient way to grant and revoke permissions to more than one user in a single statement, as well as supporting centralized administration of access.

3.1.3 Ownership and DAC Permissions

Although ASA has no concept of a database owner, it does support the concept of owners for the controlled objects (tables, views, stored procedures, user-defined functions). When a user of a database creates an object, that user becomes the object’s owner. In general, the owner of an object has all access permissions to the object regardless of explicitly granted or revoked access permissions. This access will persist as long as the applicable user remains the owner of the object.

All of the system stored procedures and system tables are owned by the users **SYS** or **dbo.** However, ASA does not allow any user to connect (i.e., login) to **SYS** or **dbo**; as a result, no user can directly update the system tables or change a system stored procedure.

3.1.4 Residual Information Protection

Databases are implemented using the file mechanism provided by the underlying operating system. Within these files, ASA creates and manages the abstractions of its controlled objects. When a database is dropped, the associated files are deleted through the operating system interface. When a new database is created, new files are created in conjunction with the creation of a new database. However, ASA is not dependent on residual information protection provided by the operating system because it provides its own mechanism internal to the database.

When a table or index is created, pages are allocated in the database. Although the data areas of these pages are not zeroed out before use, the page header information is updated whenever a new page is used for the object. At the time of page allocation, the information in the database's allocation pages, the allocation map for the object, and the page headers, are set such that only data that has been written out may be accessed.

When a table is dropped from a database, all of rows in the system tables of that database that reference the table and its associated indexes are deleted. The allocation bitmaps for all extents allocated to the table and its indexes are zeroed out, ensuring no access to those pages. Truncating a table has the effect of deleting all data rows for the table and deallocating the associated data pages and extents from the table. When rows are 'shrunk' (i.e., columns are deleted) they are rewritten in place and the other rows are moved around so as to leave no gaps. This effectively results in a truncated table that permits no access to the deleted data. When rows are 'expanded' (i.e., columns added), new rows are written and the old rows are marked as deleted and the associated space is available for reallocation. All memory segments are written before they are read.

3.2 Security audit

The ASA Server provides its own audit mechanism. This mechanism uses two audit logs. The primary audit log is the ASA transaction log file; this log is used to record auditable events that occur within the running ASA Server. The secondary audit log is the utility audit log file; this is used by ASA utility programs to record auditable events regardless of whether the ASA Server is currently running. The audit logs are protected as ASA system data.

Each audit record identifies the event type, responsible user (except for failed login attempts), data and time of the event, an indication of success or failure of the event, and other information specific to each audit event.

The general classes of auditable actions are listed below. All actions that require a role are auditable, such as those that require System Administrator or System Security Officer (i.e., any authorized administrator). Unsuccessful attempts to perform a trusted operation by an untrusted subject also result in the generation of an audit record. The auditable actions include:

- Enabling and disabling of the audit mechanism
- Successful and failed attempts to perform functions requiring DBA authority (i.e., authorized administrator actions)
- All successful and failed login attempts
- All successful and failed Data Definition Language (DDL) and Data Manipulation Language (DML) statements
- All permission checks

ASA allows authorized administrators to enable and disable the audit function as a whole. ASA also permits authorized administrators to configure audit levels to control which auditable events will be audited when audit is enabled. After events are audited, ASA permits authorized administrators to access the data using ASA utility programs. ASA also provides the ability to import the audit logs into a database, thus permitting the SQL SELECT command to be used to search and sort based on any attributes within the audit records, including user identities.

When the available audit log space (i.e., available disk space) is exhausted, the engine or server will rollback all pending transactions and fail all subsequent requests. At this point, the transaction log must be truncated in order to continue using the database.

3.3 Identification and authentication

The ASA Server provides an identification and authentication mechanism in addition to any that might be provided by the underlying operating system. In order to access the ASA Server, a login account, including a user name and password, must be created for the user. User accounts can be established as either regular user accounts or trusted user accounts. The user name, password (hashed) and an internal Server identifier are stored and protected in a system table.

To login to the Server, the user provides their user name and password to the Server. The Server hashes the password and compares the resulting value to that stored in the system table. If either the user name or password is incorrect, the login request will fail and no functions will be made available; further, the user will be provided no indication of whether the user name or password was incorrect. If the login was successful, a subject is created on behalf of the user and is represented by a unique ASA Server identifier.

The administrator is provided direction in the *Sybase ASA Database Administration Guide* [9] to define restrictions (e.g., minimum password length) to ensure that the chance of guessing a password is sufficiently small (i.e., less than one in 5 trillion).

The determination of regular vs. trusted is made through the assignment of authorities. Authorities define special privileges available to the user. There are two security relevant authorites: DBA and Resource. DBA authority offers full permissions on all objects inside the database (other than objects owned by SYS) and allows the user to grant other users the permission to create objects and execute commands within the database. Resource authority allows a user to create any kind of object within a database rather than requiring granting permissions on individual CREATE statements.

In addition to the user's identifiers and password, any groups and authorities assigned to the user are also stored in the system tables. Groups are used to simplify access control management.

ASA allows authorized administrators to define login events that can invoke user-defined stored procedures (created by the authorized administrator)[4] that are activated each time a client connects to the ASA Server. The combination of login events and Administrator-

[4] See Section 10, Validator Comments/Recommendations (Page 22) for a Validator's Comment related to this.

defined stored procedures can effectively be used to deny access to the ASA Server based on criteria defined by the authorized administrator. Among the criteria that can be configured are user identities to reject, disallowed time periods, number of failed logins, and the maximum number of current sessions the user has. When the login event causes a Administrator-defined stored procedure to be activated, it checks the applicable attributes against the defined criteria and if any of the rules match, the connection is rejected.

3.4 Security management

ASA maintains a number of special system tables, system stored procedures, and public options to control how it operates. All system tables and system stored procedures are owned by the special user designators **SYS** or **dbo**, while public options are owned by the special user designator PUBLIC. System tables provide at most read access to other users. System stored procedures generally allow execute access, but they internally restrict their own functions based on the invoking user. For example, the stored procedure used to change passwords ensures the user is either an authorized administrator or the user associated with the password to be changed. Each system stored procedure allowing management of a security functions will succeed only when invoked by an authorized administrator. The values of public options are readable by any user, but are only settable by users with DBA authority.

These system tables are used to define user accounts (including authentication data), group memberships, and authorities. Audit data is stored and protected in separate files associated with ASA (this data is accessible using ASA utility programs). Audit parameters are stored in public options. When a database is created, all of the security-related management data is restricted to authorized administrators. Access to the security data via system stored procedures is controlled by checks implemented within those procedures.

System tables store all meta-information about the database, including all security information. These tables are read-only. The information in these system tables is changed through the use of SQL DDL statements for managing the database; these tables are not modifiable with DML statements. Furthermore, any changes made to system table data will be effective the next time the data within the table is accessed (e.g., a new user connection), and will also be applied immediately to any connected user.

Access to other non-system database objects, such as databases contents (e.g., tables) is subject to the DAC Policy; any user with sufficient privilege can manage the associated access attributes. Unlike the system table data changes, changes to access attributes of database objects will be used during the *next* attempt to access that object (currently open access is not affected). Database objects are created with access restricted only to their creator, who can subsequently change the access permissions.

3.5 Protection of the TSF

ASA instantiates itself as a process within task constructs provided by the underlying operating system. It retains exclusive control of its processes and separates and differentiates client connections based on TDS or CmdSeq connections. In addition to protecting its own processes, ASA protects its shared memory and files using features provided by the underlying operating system, ensuring that the security properties of those objects do not allow access by other operating system processes. This serves to both protect ASA itself as well as to ensure that any attempts to access the database constructs implemented by ASA must be made through ASA. Furthermore, ASA has been carefully designed to offer well-defined interfaces that ensure that access to protected resources is subject to the applicable ASA security policies.

4 Assumptions

The following assumptions underlie the evaluation of the Adaptive Server Anywhere component of SQL Anywhere Studio 9:

4.1 Usage Assumptions

First and foremost, it is assumed that all authorized administrators are non-hostile, appropriately trained, and will follow the written administrative guidance they are provided. There is no assumption about the behaviour of non-administrative users.

4.2 Environmental Assumptions

A key environmental assumption is physical security. It is assumed that appropriate physical security is provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information.

With respect to the underlying operating system for the ASA server, it is assumed that there are no general-purpose computing capabilities (e.g., compilers or user applications) available, other than those services necessary for the operation, administration and support of the DBMS. It is also assumed that the IT environment protects the TOE (and its resources) and provides time stamps with at least the same degree of assurance as that claimed by the TOE.

4.3 Clarification of Scope

4.3.1 Overarching Policies

The security requirements enforced by the TOE were designed based on the following overarching security policies:

1. The users of the TOE shall be held accountable for their actions within the TOE.
2. Access controls will ensure that only those users who have been authorized to access the protected information within the TOE will be able to do so.
3. The TOE must limit the access to information in protected resources to those authorized users who have a need to know that information.
4. The TOE shall provide an authorized administrator role for secure administration of the TOE. This role shall be separate and distinct from other authorized users.

The ST classifies these are "Organizational Security Policies"; however, they are not policies imposed by the organization actually operating the TOE. Rather, they are based on policies specified in the draft Database Protection Profile and in the Controlled Access Protection Profile, and are likely to be common policies in U.S. Government installations using this product.

4.3.2 Threats Countered and Not Countered

The TOE is designed to fully or partially counter the following threats:

- That an authorized administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.
- That a process or user may take an action that results in audit data being inappropriately accessed (viewed, modified or deleted), or that prevents future audit records from being recorded, thus masking an attacker's actions.
- That an unauthorized user, process, or external IT entity may masquerade as an authorized entity to gain access to data or TOE resources.
- That a user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.
- That a malicious process or user may gain unauthorized access to the authorized administrator account, or that of other trusted personnel.
- That a malicious user or process may cause the TOE, configuration data, or sensitive user data to be inappropriately accessed (viewed, modified or deleted), allowing a breach in the TSF security policies.
- That a user may gain unauthorized access (view, modify, delete) to user data.
- That the IT operating system may fail to detect and record unauthorized actions may occur.
- That an authorized administrator may fail to identify and act upon unauthorized actions may occur.

However, users of the TOE should be cautioned that:

- The TOE *does not* counter the threat of network transmissions being observed, as packets are not encrypted for transmission. This is addressed by an assumption of physical protection. Although the SQL Anywhere Studio product does include components that encrypt transmissions, those components **are not** covered by this evaluation.

5 Architectural Information

Note: The following architectural description is based on the description presented in Part I Evaluation Technical Report for Adaptive Server Anywhere and in the Adaptive Server Anywhere Security Target.

5.1 TOE Components

The TOE consists of the following components:

- **ASA Server**. An operating system process that is running the ASA Server executable. The server is multi-threaded and may be running on several processors simultaneously.
- **Operating system files.** These are files provided by the underlying operating system and used by the ASA Server to create the database abstractions. A number of operating system files are also used by the ASA Server for configuration.
- **Administrative Programs.** These are a set of program that provide the interfaces necessary for TOE administration.

The ASA Server and the Administrative Programs run as applications on top of an operating system and depend on the services exported by the operating system to function. ASA uses operating system services for process creation and manipulation; device and file processing; shared memory creation and manipulation; provision of the network stack up through the TCP layer; and security requests such as inter-process communication. The hardware upon which the operating system runs is completely transparent to ASA; ASA sees only the operating system's user interfaces.

5.2 TOE Boundaries

Figure 5-1 illustrates the Adaptive Server Anywhere TOE and its boundaries. This figure shows that the underlying operating system and its underlying hardware are not part of the TOE. Additionally, other components of the Anywhere Studio 9 product, including any database clients provided as part of the product, are not part of the TOE.

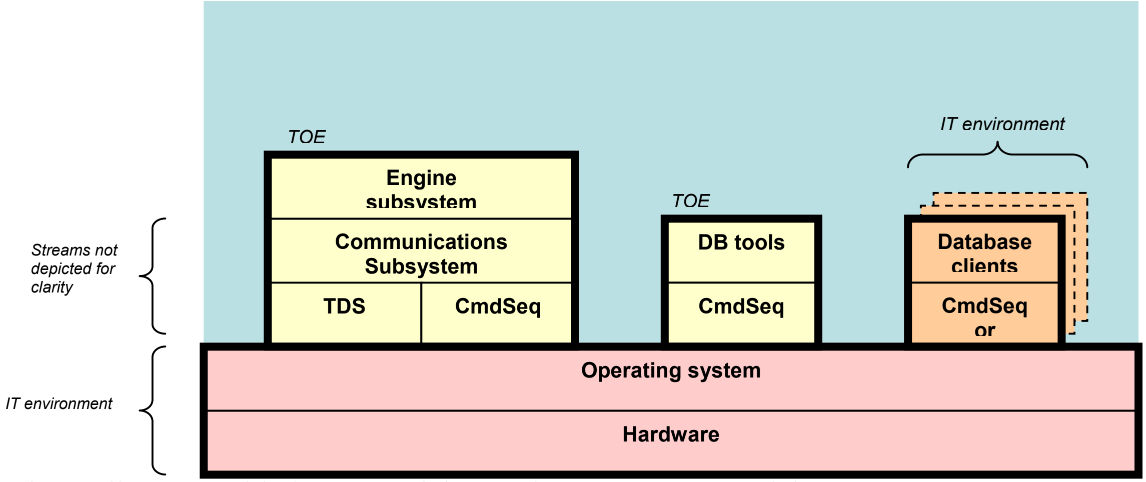

Figure 5-1. Boundaries of the Adaptive Server Anywhere TOE

There are two mechanisms available to communicate with the ASA Server: the Command Sequence protocol (CmdSeq) and the Tabular Data Stream (TDS) Protocol. These protocols are used to request ASA Server services. They are used by untrusted client processes, via routines in provided libraries, to communicate with the Server. Administrators interface with the ASA Server via utility programs provided to facilitate ASA administration. These utility programs use available library routines, just like other untrusted clients, to interface to CmdSeq and/or TDS, which in turn communicates with the ASA Server over TCP/IP or other network protocols implemented by the hosting operating system.

5.3 Architecture

ASA consists of the following distinct modules:

- **Communications subsystem**. This subsystem provides network interfaces that are accessible by client-side applications to access the query execution engine subsystem

- **Query execution engine subsystem**. This subsystem provides server-side relational database application functionality.

- **DB tools subsystem**. This subsystem provides client-side administrator console applications to manage query execution engine subsystem log files

5.3.1 Communications Subsystem

The Communications Subsystem accepts as its only input messages formatted to the specification of the CmdSeq and TDS protocols. Both protocols are used for transfer of requests and responses between clients and the Server. The Communications subsystem receives the protocol request from the client in packets. Packets of either protocol are not encrypted. The Communications Subsystem unbundles the request, and decides how the request should be resolved, based on its type. For each request, the Communications subsystem bundles the response in a packet and dispatches it to the requesting client.

5.3.2 Query Execution Engine Subsystem

When a user connects to ASA through the initiation and establishment of a CmdSeq or TDS connection, the Query execution engine subsystem activates any login trigger associated with the user after the Communications subsystem has successfully processed the login record. After a user has logged in to the Server, the Query execution engine subsystem processes SQL commands that it receives from the Communications subsystem, including:

- Queries and DML commands (select, update, insert and delete)

- Schema commands (create table, database, and others)

- Stored procedure execution, including system stored procedures
- Cursor commands
- Set statements
- Branching statements (if, while)
- Transaction control statements

The Query execution engine subsystem generates result sets, individual return values and status as a result of executing SQL statements. It returns the results to the Communications subsystem for formatting a response to the client's request, according to the CmdSeq or TDS protocol.

The SQL language restricts users from gaining access to data that has not been directly made available by the Query execution engine subsystem. The Query execution engine subsystem limits a session's access to data based on the access control policies defined for the objects.

The Query execution engine subsystem includes a kernel subcomponent that abstracts the operating-system specific services for a consistent view, regardless of the underlying operating system. The Query execution engine subsystem and the Communications subsystem use kernel services in the following areas:

- Engine Management
- Task Management
- Memory Management
- Network I/O
- Disk I/O
- Initialization, Startup and Shutdown

5.3.3 DB Tools Subsystem

The DB tools subsystem provides individual executable command-line interfaces to read audit records from the audit trail, to search and sort the audit trail, and to prevent unauthorized modifications to the audit trail by restricting access to its interfaces to administrators. The DB tools subsystem consists of the dblog, dbtran, and dbwrite executables. In the evaluated configuration, DB tools subsystem executables support the audit security function for running database servers. For example, dbtran only restricts access to its interfaces when reading from running databases, not when reading from a log file from a database that is not running.

5.4 IT Security Environment

Adaptive Server Anywhere requires an IT environment that protects the TOE (and its resources) and provides time stamps with at least the same degree of assurance as that claimed by the TOE. It is important to note that not all of the potential platforms meet the stated assumption for the IT environment. According to publicity material from the vendor, Adaptive Server Anywhere is available for a variety of Windows platforms (95/98/Me, NT, 2000, XP, XP Embedded, Tablet PC, Server 2003 64-bit Itanium and 32-bit, Mobile – Pocket PC/Handheld PC); Mac OS X; Solaris/SPARC; Linux; and has deployment options for HP-UX, HP-UX 64-bit Itanium, Linux 64-bit Itanium, IBM AIX, Compaq Tru-64. Of these, the 95/98/Me line is known not to provide appropriate domain protection. The product was tested only on Windows 2000, XP, Server 2003, Solaris, and Linux.

6 Documentation

The following documentation was used as evidence for the evaluation of the Adaptive Server Anywhere, versions 9.0.1 and 9.0.2:[5]

6.1 Design documentation

Document	Revision	Date
Adaptive Server Anywhere Architecture Summary	1.3	2005-09-21
Adaptive Server Anywhere Command Sequence Specification	0.1	2005-06-15
Adaptive Server Anywhere Security Functional Requirements	0.4	2005-05-06
SQL Correspondence	2.0	2005-09-26
TDS Correspondence	2.0	2005-06-23
TDS 5.0 Functional Specification	3.4	2005-08

6.2 Guidance documentation

Document		Revision	Date
Supplement for Installing Adaptive Server Anywhere for Common Criteria Configuration	DC00080-01-1252-01	0.1	2006-04
Sybase ASA SQL Reference	DC38129-01-0901-01	9.0.1	2004-01
	DC38129-01-0902-01	9.0.2	2004-10
Sybase ASA Error Messages	DC38131-01-0901-01	9.0.1	2004-01
	DC38131-01-0902-01	9.0.2	2004-10
Sybase ASA Database Administration Guide	DC38123-01-0901-01	9.0.1	2004-01
	DC38123-01-0902-01	9.0.2	2004-10

[5] This documentation list is based on the lists provided in the Evaluation Technical Report, Parts 1 and 2, developed by SAIC.

SQL Anywhere Studio Security Guide	DC38177-01-0901-01	9.0.1	2004-01
	DC38177-01-0902-01	9.0.2	2004-10

6.3 Configuration Management and Lifecycle documentation

Document	Revision	Date
Sybase ASA Configuration Management Plan	0.6	2006-02-16
Sybase Adaptive Server Anywhere Life Cycle Document	0.1	2004-09-28
Video Files (as documented in CC_Video_Script)	1.0	2006-02-16
iAnywhere Solutions Style Guide	1.0	2004-06
iAnywhere Solutions Authoring Guide	1.0	2003-09

6.4 Delivery and Operation documentation

Document	Revision	Date
Sybase Adaptive Server Anywhere Delivery and Operation Procedures	0.1	2004-05-26
Supplement for Installing Adaptive Server Anywhere for Common Criteria Configuration DC00080-01-1252-01	0.1	2006-04
Memorandum for Record for Sybase Adaptive Server [Anywhere|IQ], CCTL:SAIC, Review Comments, ADO Assessments (June 14, 2004), Record Number: VID4046-7-MR-0006	N/A	2004-06-14

6.5 Test documentation

Document	Revision	Date
Sybase ASA Security Function Test Documentation	0.8	2005-12-02
SQL Correspondence (test coverage)	2.0	2005-09-26
Test Scripts as referenced by Security Function Test Documentation	N/A	2006-01-20
Test Results as referenced by test scripts	N/A	2006-02-10
Sybase ADV ETR, including ADV evidence	3.0	2006-02-10

6.6 Vulnerability Assessment documentation

Document	Revision	Date
Sybase ASA Vulnerability Analysis	1.2	2006-01-10

6.7 Security Target

Document	Revision	Date
Sybase Adaptive Server Anywhere Security Target	1.0	2006-04-11

7 IT Product Testing

This section describes the testing efforts of the developer and the Evaluation Team. It is derived from information contained in the Evaluation Team Test Plan, contained in Part II of the ETR, and has been reviewed to ensure it does not contain vendor proprietary information.

7.1 Developer Testing

Evaluator analysis of the developer's test plans, test scripts, and test results indicated that the developer's testing is adequate to satisfy the requirements of EAL3.

The developer's tests were completely automated. The approach to security testing for ASA was subsystem based. To do this, the vendor developed a set of test suites that corresponded to the subsystems identified in the design documentation. Each test suite was subdivided into specific interfaces, and each test procedure targeted the specific security behavior associated with that interface.

The Sybase ASA Security Function Test Documentation included a description of the intent of tests for subsystem interface types and the testing approach. The documentation also included the test case description, the test procedures, and the expected results.

The evaluation team verified that the test coverage was suitable through analysis of the developer-provided test documentation. This analysis verified that the tests provided adequate coverage of all interfaces. Given the mapping is complete with respect to interfaces, the evaluation team concluded the coverage is complete with respect to the requirements.

With respect to depth, the evaluation team was able to trace all aspects of the implementation of security functions in the high-level design back to test cases. Multiple test cases existed for every interface, ensuring proper negative, positive, and boundary testing.

The developer provided the evaluation team with actual results for both ASA 9.0.1 and ASA 9.0.2 on the Solaris 8, Red Hat Linux 2.1, Window Server 2003 SP1, and Windows XP Professional SP2 platforms. The evaluators verified that these actual results were consistent with the expected results for each test script.

7.2 Evaluation Team Independent Testing

In addition to developer testing, the CCTL conducted its own suite of tests. The evaluation team used two platforms to perform its testing: Windows 2000 SP4 and Sun Solaris 8. The evaluation team chose these two platforms to perform its testing because the source code is different between Windows and Unix machines:

- Windows provides a common interface for all windows applications.

- Solaris was selected as a representative of the remaining Unix platforms. All of the security code on all of the Unix platforms is identical. The only source code differences among the Unix platforms are Kernel services (process handling, signal handling, network IO, disk IO, etc). The evaluation team verified this assertion through discussion with the vendor. The evaluation team confirmed with Sybase there are no compile or runtime difference between the different Unix products within the security code.

The CCTL verified that each of these platforms was running the TOE version of the firmware and the software. The CCTL installed the TOE and configured it in accordance with the provided guidance.

During its testing, the evaluation team re ran the entire vendor test suite on all test configurations. All tests were successful.

The evaluation team also developed eight (8) independent tests. The team tests developed were primarily based upon the evaluation team's analysis of the design documentation, user guidance, security target, and test documentation. Focus was placed upon areas where the developer test documentation did not cover completely. The validator reviewed these independent tests and felt that they provided sufficient supplemental coverage to the vendor tests. The evaluation team used the exact configuration documented in the vendor test documentation and used to perform the vendor test subset was used to perform the team test. The evaluation team also used the same test tools documented in the vendor test documentation to perform the team test subset.

These tests identified some discrepancies between the actual implementation and the implementation documented in the ST. The ST was updated to reflect the actual implementation.

7.3 Evaluation Team Penetration Testing

The CCTL also conducted penetration testing, using the same setup used for the independent team tests.

Prior to developing its tests, the CCTL followed well-established penetration test development procedures. This effort considered design documentation evaluation, guidance documentation evaluation, test documentation evaluation, code review, vulnerability analysis evaluation. It was revisited subsequent to the running of a portion of the vendor test subset. Therefore, it took advantage of TOE knowledge gained from each of these activities.

This resulted in a set of eight (8) penetration tests. The validator reviewed these tests, and felt that they adequately explored areas of potential vulnerability. Execution of these tests resulted in some documentation clarifications, but identified no security vulnerabilities.

8 Evaluated Configuration

The evaluated configuration, as defined in the Security Target, consists of the Sybase Adaptive Server Anywhere (version 9.0.1 or 9.0.2) component of Sybase SQL Anywhere

Studio 9, configured in accordance with the Guidance Documentation in the TOE. Specifically, the following versions of ASA were evaluated:

- Adaptive Server Anywhere version 9.0.2 build 3221 for Microsoft Windows XP, Windows 2000, and Windows 2003 Server.
- Adaptive Server Anywhere version 9.0.2 build 3219 for Sun Solaris 8, and Redhat Linux Advanced Server 2.1.
- Adaptive Server Anywhere version 9.0.1 build 2085 for Microsoft Windows XP, Windows 2000, Windows 2003 Server, Sun Solaris 8, and Redhat Linux Advanced Server 2.1.

9 Results of the Evaluation

The evaluation was conducted based upon the Common Criteria (CC), Version 2.1, dated August 1999 [1,2,3,4]; the Common Evaluation Methodology (CEM), Version 1.0, dated August 1999 [6]; and all applicable International Interpretations in effect on April 1, 2004. The evaluation confirmed that the Sybase Adaptive Server Anywhere (version 9.0.1 or 9.0.2) component of Sybase SQL Anywhere Studio 9 is compliant with the Common Criteria Version 2.1, functional requirements (Part 2) and assurance requirements (Part 3) for EAL3 augmented with ALC_FLR.2. The details of the evaluation are recorded in the CCTL's evaluation technical report, *Evaluation Technical Report for Sybase Adaptive Server Anywhere*, Part 1 (Non-Proprietary) and Part 2 (Proprietary). The product was evaluated and tested against the claims presented in the Adaptive Server Anywhere Security Target v1.0, 11 April 2006.

The validator followed the procedures outlined in the Common Criteria Evaluation Scheme publication number 3 for Technical Oversight and Validation Procedures. The validator has observed that the evaluation and all of its activities were in accordance with the Common Criteria, the Common Evaluation Methodology, and the CCEVS. The validator therefore concludes that the evaluation team's results are correct and complete.

The following evaluation results are extracted from the non-proprietary Evaluation Technical Report provided by the CCTL, and are augmented with the validator's observations thereof.

9.1 Evaluation of the Security Target (ASE)

The evaluation team applied each ASE CEM work unit. The ST evaluation ensured the ST contains a description of the environment in terms of policies and assumptions, a statement of security requirements claimed to be met by the ASE product that are consistent with the Common Criteria, and product security function descriptions that support the requirements.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.2 Evaluation of the Configuration Management Capabilities (ACM)

The evaluation team applied each EAL 3 ACM CEM work unit. The ACM evaluation ensured the TOE is identified such that the consumer is able to identify the evaluated TOE. The evaluation team ensured the adequacy of the procedures used by the developer to accept, control and track changes made to the TOE implementation, design documentation, test documentation, user and administrator guidance, security flaws and the CM documentation. To support the ACM evaluation, the evaluation team received Configuration Management (CM) records from Sybase.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.3 Evaluation of the Delivery and Operation Documents (ADO)

The evaluation team applied each EAL 3 ADO CEM work unit. The ADO evaluation ensured the adequacy of the procedures to deliver, install, and configure the TOE securely. The evaluation team ensured the procedures addressed the detection of modification while in transit. The evaluation team followed the Configuration Guide to test the installation procedures to ensure the procedures result in the evaluated configuration.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.4 Evaluation of the Development (ADV)

The evaluation team applied each EAL 3 ADV CEM work unit. The evaluation team assessed the design documentation and found it adequate to aid in understanding how the TSF provides the security functions. The design documentation consists of a functional specification and a high-level design document. The evaluation team also ensured that the correspondence analysis between the design abstractions correctly demonstrated that the lower abstraction was a correct and complete representation of the higher abstraction.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.5 Evaluation of the Guidance Documents (AGD)

The evaluation team applied each EAL 3 AGD CEM work unit. The evaluation team ensured the adequacy of the user guidance in describing how to use the operational TOE. Additionally, the evaluation team ensured the adequacy of the administrator guidance in describing how to securely administer the TOE. Both of these guides were assessed during the design and testing phases of the evaluation to ensure they were complete.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.6 Evaluation of the Life Cycle Support Activities (ALC)

The evaluation team applied each EAL 3 ALC CEM work unit. The evaluation team ensured the adequacy of the developer procedures to protect the TOE and the TOE documentation during TOE development and maintenance to reduce the risk of the introduction of TOE exploitable vulnerabilities during TOE development and maintenance. To support the ALC evaluation, the evaluation team received a video recording of the security measures at Sybase to support the documented measures.

In addition to the EAL 3 ALC CEM work units, the evaluation team applied the ALC_FLR.2 work units from the CEM supplement. The flaw remediation procedures were evaluated to ensure that flaw reporting procedures exist for managing flaws discovered in the TOE.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.7 Evaluation of the Test Documentation and the Test Activity (ATE)

The evaluation team applied each EAL 3 ATE CEM work unit. The evaluation team ensured that the TOE performed as described in the design documentation and demonstrated that the TOE enforces the TOE security functional requirements. Specifically, the evaluation team ensured that the vendor test documentation sufficiently addresses the security functions as described in the functional specification and high level design specification. The evaluation team performed a sample of the vendor test suite, and devised an independent set of team test and penetration tests. The vendor tests, team tests, and penetration tests substantiated the security functional requirements in the ST.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.8 Vulnerability Assessment Activity (AVA)

The evaluation team applied each EAL 3 AVA CEM work unit. The evaluation team ensured that the TOE does not contain exploitable flaws or weaknesses in the TOE based upon the developer strength of function analysis, the developer vulnerability analysis, the developer misuse analysis, and the evaluation team’s misuse analysis and vulnerability analysis, and the evaluation team’s performance of penetration tests.

The validator reviewed the work of the evaluation team, and found that sufficient evidence and justification was provided by the evaluation team to confirm that the evaluation was conducted in accordance with the requirements of the CEM, and that the conclusion reached by the evaluation team was justified.

9.9 Summary of Evaluation Results

The evaluation team's assessment of the evaluation evidence demonstrates that the claims in the ST are met. Additionally, the evaluation team's performance of a subset of the vendor tests suite, the independent tests, and the penetration test also demonstrated the accuracy of the claims in the ST.

The validation team's assessment of the evidence provided by the evaluation team is that it demonstrates that the evaluation team followed the procedures defined in the CEM, and correctly verified that the product meets the claims in the ST.

10 Validator Comments/Recommendations

- The ST claims that the IT environment defines an environment requiring more security than the U.S. Government Protection Profile Consistency Guidance for Basic Robustness, dated 24 July 2002, which is claimed to be comparable to or better than the historical notion of the C2 level of the Trusted Computer System Evaluation Criteria. Although this claim is true with respect to assurance components, the issue of "more security" is more nebulous, as that includes functionality and threat coverage as well.
- In the ST, the requirements on the IT Environment are written as if they were levied on the TSF. This is confusing to the reader. Readers of the ST should liberally interpret these requirements in the context of the IT environment.
- Although the installation CDs are uniquely identified, the specific sub-versions of ASA are not clear on the SQL Anywhere Studio packaging. Users of this product are cautioned to check the version on the CD before proceeding with installation.
- When delivered as a physical package, the TOE provides no information regarding the Common Criteria configuration; in fact, the documentation CDs contain the security guide for the prior C2 evaluation of the product. Users of the TOE are reminded to obtain the correct configuration information from the URL specified in the VPL entry before installation, and to ensure they have the latest version of the security guidance on the product before using the product.
- The *SQL Anywhere Studio Security Guide* [10] talks primarily about the C2 certification, noting:

  > Adaptive Server Anywhere version 7.0 achieved the C2 security certification of the US federal government. The C2 section of this manual describes how to operate the current version of Adaptive Server Anywhere in a manner comparable to the C2-certified configuration.

> This book is not the certified document describing C2 compliance. The certified documentation is available from the Sybase Web site at http://www.sybase.com/detail?id=1010458. Nothing in this document should be taken to suggest that the current version of the software is C2 compliant. Use of the phrase "equivalent to the C2-certified configuration" and similar phrases does not imply actual C2 compliance. The only way to operate in a C2-certified manner is to use the C2-certified release of the software according to the C2-certified documentation.

This is misleading to the consumer, as Version 7 of the product is no longer available and the C2 rating has been deprecated. The validator hopes that, in the future, the vendor will update this document to refer to the Common Criteria validated product.

- The SFRs FIA_AFL.1, FTA_MCS_EXP.1, and FTA_TSE.1 are implemented via stored procedures manually created by the administrator during the installation process based on material provided in the installation manual. The administrator is cautioned to cut-and-paste this text from the electronic version of the manual; *it should not be manually copied and hand-entered*. Only after the text is pasted should the specific configuration parameters be modified. Failure to do this may result in incorrect enforcement of the indicated SFRs.
- The behaviour of the system for authentication failures is atypical. Most systems reset the failure count after a specific amount of time; for example, you are locked out for 30 minutes after three login failures. This is not the case for Adaptive Server Anywhere: the count of failures continues to increment until reset by an administrator.

11 Annexes

Not applicable.

12 Security Target

The Security Target is identified as *Sybase Adaptive Server Anywhere Security Target,* Version 1.0, 11 April 2006.

13 Glossary

The following definitions are used throughout this document:

- **Adaptive Server Anywhere (ASA).** The relational database server component of SQL Anywhere Studio, intended for use in mobile and embedded environments or as a server for small and medium-sized businesses.
- **Authentication.** Verification of the identity of a user.

- **Command Sequence (CmdSeq).** A protocol used to direct the actions of the ASA server.
- **Common Criteria Testing Laboratory (CCTL)**. An IT security evaluation facility accredited by the National Voluntary Laboratory Accreditation Program (NVLAP) and approved by the CCEVS Validation Body to conduct Common Criteria-based evaluations.
- **Conformance**. The ability to demonstrate in an unambiguous way that a given implementation is correct with respect to the formal model.
- **Data Definition Language (DDL).** The subset of SQL statements for modeling the structure of a database. DDL statements create, modify, and remove database objects, including users.
- **Data Manipulation Language (DML).** The subset of SQL statements for retrieving and updating the contents of a database.
- **Database.** A collection of tables that are related by primary and foreign keys. The tables hold the information in the database. The tables and keys together define the structure of the database. A database-management system accesses this information.
- **Database Administrator (DBA).** The user with the permissions required to maintain the database. The DBA is generally responsible for all changes to a database schema, and for managing users and user groups. The role of database administrator is automatically built into databases as user ID DBA with password SQL.
- **Database File.** A database is held in one or more database files. There is an initial file, and subsequent files are called dbspaces. Each table, including its indexes, must be contained within a single database file.
- **DBA Authority.** The level of permission that enables a user to carry out administrative activity in the database. The DBA user has DBA authority by default.
- **Database Owner (dbo)**. A special user that owns the system objects not owned by SYS
- **Evaluation**. The assessment of an IT product against the Common Criteria using the Common Criteria Evaluation Methodology to determine whether or not the claims made are justified; or the assessment of a protection profile against the Common Criteria using the Common Evaluation Methodology to determine if the Profile is complete, consistent, technically sound and hence suitable for use as a statement of requirements for one or more TOEs that may be evaluated.
- **Evaluation Evidence**. Any tangible resource (information) required from the sponsor or developer by the evaluator to perform one or more evaluation activities.
- **Grant Option.** The level of permission that allows a user to grant permissions to other users.
- **Group.** A special user identity that is allowed to have members.

- **Local Temporary Table.** A type of temporary table that exists only for the duration of a compound statement or until the end of the connection.
- **Log File.** A log of transactions maintained by Adaptive Server Anywhere. The log file is used to ensure that the database is recoverable in the event of a system or media failure, to improve database performance, and to allow data replication.
- **Metadata.** Data about data. Metadata describes the nature and content of other data.
- **MobiLink**. A session-based synchronization technology designed to synchronize UltraLite and Adaptive Server Anywhere databases with many industry-standard SQL database-management systems from Sybase and other vendors. **MobiLink is not covered by this evaluation**.
- **Relational Database-Management System (RDBMS).** A type of database-management system that stores data in the form of related tables.
- **Schema**. The structure of a database, including tables, columns, and indexes, and the relationships between them.
- **SQL.** The language used to communicate with relational databases. ANSI has defined standards for SQL, the latest of which is SQL-99 (also called SQL3). SQL stands, unofficially, for Structured Query Language.
- **SQL Remote.** A message-based replication technology for two-way replication between consolidated and remote databases. The consolidated database must be Adaptive Server Anywhere or Adaptive Server Enterprise. The remote databases must be Adaptive Server Anywhere. **SQL Remote is not covered by this evaluation.**
- **Stored Procedure.** A program comprised of a sequence of SQL instructions, stored in the database and used to perform a particular task.
- **SYS.** A special user that owns most of the system objects. You cannot log in as SYS.
- **System Object.** Database objects owned by SYS or dbo (Data Base Owner).
- **System Table.** A table, owned by SYS or dbo, that holds metadata. System tables, also known as data dictionary tables, are created and maintained by the database server.
- **System View.** A type of view, included in every database, that presents the information held in the system tables in an easily understood format.
- **Tabular Data Stream (TDS).** A protocol used to direct the actions of the ASA server.
- **Target of Evaluation (TOE)**. A group of IT products configured as an IT system, or an IT product, and associated documentation that is the subject of a security evaluation under the CC.
- **Temporary Table.** A table that is created for the temporary storage of data. There are two types: global and local.
- **Transaction.** A sequence of SQL statements that comprise a logical unit of work.

- **Transaction Log.** A file storing all changes made to a database, in the order in which they are made. It improves performance and allows data recovery in the event the database file is damaged.
- **UltraLite.** A deployment technology for Adaptive Server Anywhere databases, aimed at small, mobile, and embedded devices. Intended platforms include cell phones, pagers, and personal organizers. **UltraLite is not covered by this evaluation.**
- **Validation**. The process carried out by the CCEVS Validation Body leading to the issue of a Common Criteria certificate.
- **Validation Body**. A governmental organization responsible for carrying out validation and for overseeing the day-to-day operation of the NIAP Common Criteria Evaluation and Validation Scheme.
- **View.** A SELECT statement that is stored in the database as an object. It allows users to see a subset of rows or columns from one or more tables. Each time a user uses a view of a particular table, or combination of tables, it is recomputed from the information stored in those tables. Views are useful for security purposes, and to tailor the appearance of database information to make data access straightforward.

14 Bibliography

The Validation Team used the following documents to produce this Validation Report:

[1] Common Criteria Project Sponsoring Organisations. *Common Criteria for Information Technology Security Evaluation: Part 1: Introduction and General Model*, Version 2.1, August 1999. CCIMB-99-031.

[2] Common Criteria Project Sponsoring Organisations. *Common Criteria for Information Technology Security Evaluation: Part 2: Security Functional Requirements*, Version 2.1, August 1999. CCIMB-99-032.

[3] Common Criteria Project Sponsoring Organisations. *Common Criteria for Information Technology Security Evaluation: Part 3: Security Assurance Requirements*, Version 2.1, August 1999. CCIMB-99-033.

[4] Common Criteria Project Sponsoring Organisations. *Common Evaluation Methodology for Information Technology Security* – Part 1: Introduction and general model, Version 0.6, 11 January 1997.

[5] Common Criteria Project Sponsoring Organisations. *Common Evaluation Methodology for Information Technology Security* – Part 2: Evaluation Methodology, Version 1.0, August 1999.

[6] Common Criteria, Evaluation and Validation Scheme for Information Technology Security, *Guidance to Validators of IT Security Evaluations*, Scheme Publication #3, Version 1.0, January 2002.

[7] iAnywhere Solutions (A Sybase Company). *Introducing SQL Anywhere Studio.* Version 9.0.2. DC38120-01-0902-01. October 2004. Available at

http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/en/html/dbfgen9/dbfgen9.htm.

[8] iAnywhere Solutions (A Sybase Company). *Supplement for Installing Adaptive Server Anywhere for Common Criteria Configuration*, DC00080-01-1252-01, April 2006. Available at http://www.ianywhere.com/developer/product_manuals/sqlanywhere/sqlanywhere_cc_configuration.pdf.

[9] iAnywhere Solutions (A Sybase Company). *ASA Database Administration Guide*. DC38123-01-0902-01, October 2004. Version 9.0.2. Available at http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/en/html/dbdaen9/dbdaen9.htm.

[10] iAnywhere Solutions (A Sybase Company). *SQL Anywhere Studio Security Guide*. Version 9.0.2. DC38177-01-0902-01, October 2004. Available at: http://www.ianywhere.com/developer/product_manuals/sqlanywhere/0902/en/html/dbseen9/dbseen9.htm

[11] Science Applications International Corporation. *Adaptive Server Anywhere Security Target,* Version 1.0, April 11, 2006

[12] Science Applications International Corporation. · *Evaluation Technical Report for the Sybase Adaptive Server Anywhere, Versions 9.0.1 and 9.0.2, Part I (Non-Proprietary)*, Version 7.0, 18 April 2006

[13] Science Applications International Corporation. · *Evaluation Team Test Plan for the Sybase Adaptive Server Anywhere, ETR Part II Supplement (SAIC and Sybase Proprietary)*, Version 2.0, 17 February 2006.

Note: This document was used only to obtain the description of the test effort.