

GUIDE **TO** **SYBASE SECURITY**

Author: Nilesh Burghate
nileshb@nii.co.in

Company: Network Intelligence India Pvt. Ltd.
<http://www.nii.co.in>

Date: 31st January 2003

Guide to Sybase Security

Introduction:

This article provides a detailed explanation of security for a Sybase database. The article has been written from the perspective of both security auditing and implementation. The queries and details given have been tested on Sybase Adaptive Server Enterprise 12.5 but will be valid for other versions as well.

Sybase Server provides good security mechanisms, if configured properly. The security Architecture implemented by Sybase can be divided into five parts:

1. **Dictionary Access Controls (DAC):** This helps database administrators to give access to users at the object level.
2. **Identification and Authentication controls:** Ensures only authorized users have access to objects.
3. **Division or Roles:** The divisions are based on system and user level tasks. There is a special role, 'sa_role' for Database and System Administration, and 'sso_role' for Security Officer who is responsible for managing security of database.
4. **Network based Security:** Sybase provides protected Remote Authentication, encrypted transmission of data over networks, and message integrity.
5. **Auditing:** Sybase has very good auditing mechanisms. If configured to its potential it provides for excellent accountability.

While carrying out the following tasks, you should have both 'sso_role' and 'sa_role' roles. Also, for all stored procedures use the *master* database by issuing the 'use master' query. You may bunch all the queries into a single file called QueryFile.txt and issue the commands at one go. To save the output of the queries into an output file you may issue the following from the command line

```
isql -S ServerName -U UserName -i QueryFile.txt -o Output.txt
```

We cover the following topics within Sybase Security:

- General Server Security
- Database Configuration
- User Security
- Data Security
- Auditing
- Network Security

1. General Server Security

1.1 Operating System Check

- Check for permissions on the directory where Sybase is installed. Ensure only System Administrator has access to Sybase directory.
- Using regedt32 Check for permissions on Sybase Registry Keys on HKLM\Software\Sybase

1.2 Server Information

- To list Remote servers, Backup servers and other Servers on which this Adaptive Server can execute RPC, along with the server options:
exec sp_helpserver

In the above output check for the following parameters:

Network password encryption:

If "*net password encryption*" is true, the client-server authentication is done by encrypting passwords using the key supplied by server.

If "*net password encryption*" is false, the client-server authentication takes place in plain text.

Security mechanism:

By default, "*rpc security model A*" is used, which does not provide security mechanisms such as encryption

For securing network communication, the RPC security mechanism must be set to "*rpc security model B*" which provides various security services such as Mutual Authentication, Message Confidentiality via Encryption, and Message Integrity.

- To get information about the databases present on a particular server:
exec sp_helpdb

1.3 Login Configuration

- Authentication Modes:
Check which authentication mode is enabled:
exec sp_loginconfig "login mode"

If the value is:

0 – Standard

1 – Integrated

2 – Mixed mode

Standard Authentication Mode:

This is the default Authentication mode. Sybase uses its own database (the table *syslogins* in the master database) to authenticate users. Even a Windows NT/2000 Administrator cannot log into the Sybase Server if he has no valid login account in the Sybase server.

Integrated Authentication Mode:

In Integrated Authentication Mode, Sybase relies solely on Windows to authenticate users. Windows users or groups are then granted access to Sybase.

Mixed Mode Authentication:

In Mixed Mode, users are first authenticated by Windows Authentication and if the valid Windows account is not present, Sybase uses its own database i.e. the Standard Security mode to authenticate users who are authenticated by Sybase username-password pairs maintained within the Sybase Server.

- Default Login:
exec sp_loginconfig "default account"

Ensure that the value of the parameter *DefaultLogin* is not "sa". This parameter is used when Sybase is configured for Integrated Security mode. When the authorized user does not have any entry in *syslogins* table, Sybase allows him to enter with *DefaultLogin*'s credentials. Therefore any authorized Windows Users will be became "sa", if *DefaultLogin* value is set to "sa". It should ideally be set to NULL or a low-privileged account.

1.4 Patch Levels

The server must be kept updated with the latest patches. To get the detail of current version and patches applied:
select @@VERSION

The latest patches can be downloaded from <http://downloads.sybase.com/swd/swx>
Some of the terminologies used for Sybase patches:

- **EBF (Expedited Bug Fix or Emergency Bug Fix):** this is similar to hotfixes for Windows. Sybase Team releases EBFs for the Bugs found in the software. Previously these were called SWR (Software Release).
- **ESD (Electronic Software Delivery):** these are similar to Service Packs. These are Roll Ups of EBF's that were made available for downloads.
- **IR (Interim Release):** These are Super Roll Ups of ESD's which also contain some of the Extra Features added to the Sybase Server.

Note: You must apply the previous IR before applying latest ESD. Read the instructions carefully before applying patches

2. Database Configuration

2.1 General Database Parameters

- To get current configuration for the Sybase ASE Server:
exec sp_configure
- Check for 'allow updates to system tables'. DBAs can write stored procedures to modify system tables if 'allow updates to system tables' is 'on'. Therefore it is the duty of System Security Officer to ensure this key is set to 'off'.
exec sp_configure "allow updates to system tables"
If the stored procedure has already been created to modify system tables, it is possible to execute this stored procedure even if the current value of 'allow updates to system tables' is set to 'off'. Therefore, it is very important to audit the list of stored procedures on the database.
- Check for the key 'allow resource limit', and ensure it is set to '1'
exec sp_configure "allow resource limit"
- Sybase allows limiting of resources (number of rows to output, query processing time, and bulk inserts) used by a particular User or Application. This information is stored in the table *sysresourcelimit*. Malicious users can cause DoS attacks by executing a query, which takes long time to execute. Setting this key to '1' and making proper configurations for users and applications, it is possible to prevent such attacks.
- Ensure that system table '*syscomments*' is protected
exec sp_configure "select on syscomments.text"
- The system table *syscomments* contains the text of views, triggers, default table constraints, and procedures. This data is very critical and should be protected, by denying select permission on this table. By default, the value is set to '1', which allows select permission on this table. Ensure that this value is set to '0'.

2.2 Error Log Configuration

- Check if failed login attempts are logged to error logs
exec sp_configure "log audit logon failure"
Ensure this value is '1', which would log all failed login attempts to error logs in addition to the audit tables, which help in recording such critical intrusion attempts at two different places. If this value is '0', failed login attempts are not logged to error logs.
- Check if successful login attempts are logged to error logs
exec sp_configure "log audit logon success"
Ensure this value is '1'.

3. User Level Security

3.1 Group Details

- To get all groups present in the Particular Database
use DBName
exec sp_helpgroup
- To get a group-wise listing of users present in each Database (DBName)
use DBName
exec sp_helpgroup GroupName

(Repeat this for each of the group names obtained from the previous query)

3.2 Role Details

- Check for Server Roles and User-defined roles present on server
select name, password, pwdate, status from sysserverroles
- To obtain detailed information of each role:
exec sp_displayroles "RoleName", expand_up

(Note: Use *expand_up* and *expand_down* to get roles hierarchy)
- To get details about roles of a particular user
exec sp_displayroles UserName, expand_down
- Check all roles without a password
Sybase provides a mechanism where roles can be set to have passwords. This is a strong security mechanism, which prevents user from using these roles even if they are granted these roles. This facility can also be used to audit who had logged in with this role. So it is recommended to set passwords to these roles:
select name from sysserverroles where password = NULL

3.3 User Details

- To get all users in particular database:
exec sp_helpuser
- To get the hashed user passwords
select name, password from syslogins
- To get detailed information about a particular user
sp_displaylogin UserName

Guide to Sybase Security

This will produce a lot of information about that user. We must check for the following parameters:

Default login name

The login name might be different from the user name.

Default database

Ensure that no user is assigned default database as *master*, since database *master* stores all system tables, no user should be given access to this database except to users who has '*sa_role*' and '*sso_role*'.

Auto login script

If Auto login script is set for any user, check this script in details and ensure it does not contain any malicious code. This script gets executed as soon as user logs into the server.

Roles assigned

In Sybase, privileges are assigned to roles and users are assigned these roles. Therefore, you must determine what privilege levels each user has.

Whether this account locked

If the account is locked, make a note and ask DBA for reason behind it.

Last date of password change

If the date for last password change is quite old, recommend changing of password for that user.

Password expiration interval

This is the maximum validity period for the user's password. Should be between 4 weeks to 6 weeks depending upon user's privileges and the criticality of the database.

Whether password got expired?

Whether the current password of the user is expired. This might mean that this is a user who has not logged in for quite some time.

Minimum password length

This must be set according to the organization's security policy. Anywhere from 6 upwards is a good value.

Maximum failed logins

Ensure that the maximum allowed failed login attempts is not more than 3. After 3 wrong attempts his account will be locked out.

Current failed login attempts

This is critical if the 'Maximum Failed Logins' parameter has not been set. It will help you to determine if any accounts are being attacked.

- To get information about the permissions assigned to users
use DBName
exec sp_helprotect
Execute this query for each database.

Note down those users who have important permissions. The WITH GRANT is a very critical permission, and you must ensure that only legitimate users have this permission. This permission allows the user to grant his permissions to other users, and it can create a loss of accountability. DELETE and UPDATE permissions should also be reviewed for users owning these. Any user can use permissions given to 'public'. Hence ensure that the 'public' account does not have DELETE and UPDATE permissions.

3.4 Additional User Security Checks

- The 'sa' login has super-user privileges on a Sybase server. Since this account is the primary target of all attackers, it should be protected with a strong password.
- There is a better way of securing the 'sa' account however. Remove all the important roles 'sa_role' and 'sso_role' from this account, so that even if this account gets compromised Sybase database is safe from malicious attacks. Instead grant these roles to only those logins who are responsible for database administration. Grant the 'sso_role' to user who has the responsibility of System Security Officer and the 'sa_role' to the DBA. Make sure to set passwords for these roles.
- Remove the user 'Guest' from all databases except *master* and *tempdb*
- Set permissions to objects at the group level.

3.5 Password Parameters

- Check for System-wide password expiration interval
This value will expire the password of the logins when the expiration interval is over. Ensure this value is between 2-3 weeks.
 '0' - Password does not expired
 'n' - Password expires after specified 'n' number days.
exec sp_configure "password expiration interval"
- Check if password contains at least one digit
By setting this value to '1', Sybase enforces users to have at least one number in their password. This helps in preventing dictionary attacks to crack passwords.
exec sp_configure "check password for digit"
- Check for the server wide minimum password length
Minimum password length should be at least 8, which would make it difficult for users to choose easily guessable passwords.
exec sp_configure "minimum password length"

Guide to Sybase Security

- Review the passwords of the following system accounts:

User name	Default Password	Comments
mon_user	mon_user	Default Monitor Server Account
sybmail	User defined	Created when the Sybase Mail Service is installed
dba	SQL	Created with Enterprise Portal Express Edition
entldbdbo	dbopswd	Created with Database Access Control
entldbreader	rdrpswd	Created with Database Access Control
jagadmin	'NULL Password'	Created with Enterprise Portal Application Server
PIAdmin	PIAdmin	Created with Enterprise Portal Application Server
pkiuser	pkipasswd	Enterprise Portal
PortalAdmin	sybase	Enterprise Portal
pso	123qwe	Enterprise Portal

Delete system accounts, which are not required.

4. Data Level Security

4.1 Permissions

- Check for permissions on critical Tables, Procedures, Triggers
use DBName
exec sp_helprotect ObjectName

This will output:

1. Who has what permissions
 2. Type of permission depending on object
 3. Whether the WITH GRANT Permission is set or not
- Review if any objects have permissions granted to group 'public'.

4.2 Stored Procedures

- List all extended stored procedures, which are stored in *sybsysdatabase*
use sybssystemprocs
select name from sysobjects where type='XP'
- Ensure that the extended stored procedure *xp_cmdshell* is removed.
xp_cmdshell, is a very critical procedure which allows execution of Operating System commands. Default value for the security context of *xp_cmdshell* is 1, which requires user to have Windows NT account to execute OS commands, But once Sybase ASE gets compromised, the intruder can set security context of *xp_cmdsell* to 0, which allows Sybase to execute OS commands under the security context of Sybase windows NT account, thus your whole OS is at stake. So it is strongly recommended to remove this procedure.

To drop this extended procedure:

```
exec sp_dropextendedproc xp_cmdshell
```

Note: Make sure to delete the *sybsyesp.dll*. If this *.dll* is not deleted, the DBA on the server can restore *xp_cmdshell*.

However, if you need to keep this procedure then check who has permissions on *xp_cmdshell*:

```
use sybssystemprocs  
exec sp_helprotect "xp_cmdshell"
```

- Check for other extended stored procedures like *sendmail*, *freemail*, *readmail*, *deletemail*, *startmail*, *stopmail*. If you are not using the mailing facility provided by Sybase, it's recommended to remove these procedures and the mail account '*sybmail*'.

5. Auditing

Sybase does not install the auditing facility by default. This must be installed in addition to the default installation. Sybase auditing tables and procedures are located in the *sybsecurity* database, which is created when Auditing is installed on the Server.

- Check if the Sybase inbuilt Auditing facility is installed or not by ensuring that the *sybsecurity* database exists. Sybase comes with full-featured inbuilt auditing mechanism. When installed Sybase stores the auditing configuration details and audit trail tables *sysaudits_01-sysaudits_08* in *sybsecurity*.
- Check if Auditing is enabled/disabled
use master
exec sp_configure "auditing"

The following sections apply only if auditing has been installed and enabled.

- Check for number of auditing tables in *sybsecurity* database
Sybase recommends using at least 2-3 tables for Auditing. So that if one gets filled up, others can be use instantaneously without loss of data.
select count() from sysobjects where name like "sysaudits%"*
- Check if thresholds are added for any audit table and what threshold procedures are associated with each of them.
use sybsecurity
exec sp_helpthreshold aud_seg1
Repeat this query for each audit table. The segment name for "*sysaudits_01*" is "*aud_seg1*", for "*sysaudits_02*" its "*aud_seg2*", and so on.
- Check if audit tables are archived or not
It is recommended to archive the audit tables for investigating malicious activities. Determine the stored procedure associated with the particular audit table in the above query. This stored procedure gets executed when the table reaches its threshold limited. It is recommended that the SP should archive the audit table.
- Check for current audit table parameter
use master
exec sp_configure "current audit table"

The output will be either an integer 'n' or '0'. Recommended value is '0', which indicates if the current audit table is full, Sybase will use the next audit table.

- Check for the parameter Suspend Audit when audit table is full.
exec sp_configure "suspend audit when device full"

Ensure the value is "1", which will suspend auditing when all audit tables are full. This condition will arise only if the threshold procedures did not run successfully. However setting this value to "0" will truncate the current audit table and continue using it as the current audit table. Therefore if audit tables are to be archived and thresholds are used, keep this value to "1"

- Check for the key "audit queue size" and ensure it is set to large value (keep it roughly to '50')
use master
exec sp_configure "audit queue size"

A trade-off has to be maintained between security and performance, while setting this key value. A lower value will increase frequency of writing from queue to audit tables at the cost of performance and a higher value will cost security problem if the system crashes for it will result in the loss of data from queue.

- Check for the current auditing options for the following important global settings:
 - Logins - All login attempts
 - Logouts - All logout attempts
 - bcp - bulk copy event
 - create - All create events like create table, procedure, view, trigger, etc.
 - delete - Deleting of rows from tables and views.
 - disk - execution of disk init, disk reinit, etc.
 - drop - Drop events for database, tables, procedure, triggers, views, etc.
 - dump - Dumping of database, transactions, etc.*use sybsecurity*
exec sp_displayaudit
Ensure that all these setting are turned "on".

- Check for failed logins attempts
use sybsecurity
*select * from AuditTable where event =45 and eventmod =2*

For event details in audit table refer to:

<http://manuals.sybase.com/onlinebooks/group-as/asg1200e/asesag>

6. Network Level Security

6.1 Remote Server Information

- Check if Remote Server Access is allowed

use master

exec sp_configure "allow remote access"

'1' - Remote access is allowed

'0' - Remote access is not allowed

If Remote access is allowed, check for the Remote User credentials and Network Security. This allows stored procedures on the local server to be run from a remote server through RPC.

- Check for the presence of trusted user

exec sp_helpremoteserver

Sybase allows trusted connection to be set up for remote logins such that they do not need to provide a password for connecting to the server. If such trusted connections are present investigate the credentials of those users.

6.2 Remote Access Mechanism

- Check for Security mechanism and Security Services provided by them

*select * from syssecmechs*

The table *syssecmechs* is not present by default, and gets created when queried for it. It contains the following columns:

sec_mech_name Name of Security mechanism provided by the server

available_service Security services provided by that mechanism.

For instance, for Windows LAN Manager, the entries would be:

sec_mech_name = NT LAN MANAGER

available_service = unified login

- Review the *libtcl.cfg* file, which contains information about Network driver, Security, Directory drivers and any required initialization information. Check the following parameters:

1. If LDAP passwords are encrypted

2. Security mechanisms:

"dce" For the DCE security mechanism.

"csfkrb5" For the CyberSAFE Kerberos security mechanism.

"LIBSMSSP" For Windows LAN Manager on Windows NT or Windows 95(clients only).

Note: Location of *libtcl.cfg*:

UNIX platforms: \$SYBASE/config/

Desktop platforms: SYBASE_home\ini\

Guide to Sybase Security

- Check for parameter Unified login required
exec sp_configure "unified login required"
Ensure it is set to '1', if Network security is enabled. Setting it to '0', will allow traditional user-password logins to ASE as well as trusted connections, which would nullify our efforts of putting network security.