# Global Information Assurance Certification Paper

**Unix Security Assessment**
**IN-TOUCH Customer Tracking Application (CTA)**

Sheldon Brown
November 18, 2000

## Table of Contents

## Executive Summary

This security assessment examined the UNIX security on the production database server for the CTA system at IN-TOUCH (I-T). This database server is a single-purpose server that only hosts the Customer Tracking Application (CTA) database. The CTA system is considered business-critical.

This assessment looked at the following areas:

- Access control
- Directories and Files
- Physical environment
- Privileged environment
- Operating system vulnerabilities
- System administration practices
- System configuration
- System monitoring
- Application and Third-Party Software

This assessment is part of the entire IN-TOUCH security assessment. The other parts looked at the security of the Sybase database and its related software, the CTA application itself, and the network architecture.

The current state of the system security is not consistent with the desires of management. The Summary Table lists the findings and a brief summary of the corrective action. It is not any large exposure, but a large number of smaller ones.

Many of the security vulnerabilities are due to the system being put into production only recently.

## Summary Table

This summary table lists the findings and a brief summary of the corrective action that should be taken. The risk is estimated on a 0 - 10 scale, with 0 being a very small risk, and 10 a very large risk. The effort to correct uses 0 for very simple, and 10 for very difficult. The Suggested Correction Order column orders the corrections, based on the risk and estimated effort. The body of the report goes into a more detailed discussion.

| Area | Finding | Correction | Risk | Effort | Suggest. Corr. Order |
|---|---|---|---|---|---|
| **Access Control** | | | | | |
| | Poor password choices | Change passwords | 9 | 2 | 1 |
| | Passwords not changed | Develop and conform to policy | 8 | 2 | 2 |
| | Partially disabled user-ids | Implement the "noshell" tool or Completely disable them | 2 | 2 | 13 |
| | Descriptive user-ids | Change user-ids | 1 | 4 | 16 |
| **Directories & Files** | | | | | |
| | World-writable directories | Change the permissions and/or ownership | 5 | 4 | 12 |
| | World-readable security/audit directories and files | Change permissions | 4 | 2 | 8 |
| | World-writable directories referenced during system initialization | Research with Sun | ? | 2 | 15 |
| | Ownership Group Undefined | Create the group | 1 | 1 | 18 |
| **Physical Environment** | | | | | |
| | System is not in a controlled environment | Move to secured computer room | 6 | 3 | 4 |
| **Privileged Environment** | | | | | |
| | Privileged scripts found | Research, delete, or modify | 6 | 3 | 11 |

| | | | | | |
|---|---|---|---|---|---|
| **System Admin. Practices** | | | | | |
| | Security fixes not up-to-date | Install security fixes as soon as they are stable | 8 | 4 | 7 |
| | Marginal Change Control | Develop a source control/versioning Management should sign off on all changes | 3 | 6 | 9 |
| **System Configuration** | | | | | |
| | Unneeded Services Available | Disable the unneeded services | 8 | 2 | 6 |
| | UUCP is installed | Remove packages SUNWbnuu and SUNWbnur | 5 | 5 | 18 |
| **System Monitoring** | | | | | |
| | No Regular Logging and Exception Reporting | Develop and run exception reports | 4 | 3 | 5 |
| | No change/intrusion detection process used | Implement Tripwire | 5 | 7 | 10 |
| **Application and Third-Party Software** | | | | | |
| | Sybase security fixes not up-to-date | Install Sybase fixes | 2 | 2 | 14 |
| **Application Data and Processing** | | | | | |
| | None | N/A | - | - | - |
| **Backup and Disaster Preparedness** | | | | | |
| | Infrequent backups | Do scheduled, frequent backups | 7 | 1 | 3 |
| | Backups not stored off-site | Store backups off-site | 4 | 1 | 17 |

## Introduction

### Historical Background

The CTA system was procured and installed in 1999, following a recommendation that many separate ad-hoc databases on multiple PCs were not meeting the need of management for integrated information. These separate systems were also causing operational problems because of unsynchronized redundant data. CTA is the first step toward an integrated data processing system.

### Overview of the Existing Environment

The CTA server is a Sun Ultra 10. The CTA system is running the following software

| Vendor | Package | Version | Function |
|---|---|---|---|
| Sun | Solaris | 8 | Unix operating system |
| Sybase | Adaptive Server Enterprise | 12.0 | Database management system |

This system is not a mail server, web server, or file server.

Current security implementation is running the Solaris 2.8 default security.

### Desired Security Level

IN-TOUCH management, in concurrence with the System Administrator responsible for CTA, wants the system to be secure from internal and external threats.

### The Security Assessment

#### Purpose

This security assessment examined the CTA system to see if its security was consistent with the security desired by IN-TOUCH management and staff. This assessment is also serving as a prototype for security assessments of the other UNIX systems that IN-TOUCH will acquire in the near future.

#### Scope of Assessment

This portion of the assessment looked primarily at physical environment and Unix issues. This assessment did not look at network/LAN issues, or staffing and organizational issues. Application and third party software were only reviewed at a very high level. Virus protection was not considered since this system is not a server for Windows' systems.

#### Conduct and Methodology

This assessment used the following methods:

- Interviews of IN-TOUCH management and staff. Staff included the ISO, the CTA System Administrator, and the CTA Database Administrator.
- Running various security assessment tools such as COPS and NMAP.
- Observation

**Format**

The overall security is broken down into a number of areas, practices, and findings.

Each area will be discussed with the following content:

Description: This briefly describes what this area encompasses.

Discussion: This describes what this area of the assessment looked at and what was found.

Methods Used: This describes the methods and tools used to assess this area.

Within each area, there will be a number of findings, and/or some satisfactory practices.

The Satisfactory Practices area is usually included only if there is a practice that is not obvious. The Satisfactory Practices will have the following content:

Satisfactory Practices: A brief description of the practice.

Each finding will have with the following content:

Finding: A title for the deficiency

Ranking: A ranking giving an estimate of how big the risk is, how difficult it will be to correct, and a suggested order for the correction. See the Summary Table for details.

Discussion: The discussion is narrative description of the finding and its security implications.

Occurrences: The Occurrences give specific information, when appropriate. This is useful primarily or technical management and staff.

Correction: The correction is a brief narrative of how to correct the finding.

**Tool Use**

There is an appendix for each major tool used. It includes a description of the tool. The results generated by the tool are summarized. If there were false positives (in this context being security issues flagged that were not, in fact, security issues), they are listed and explained. Please note that these are false positives on this particular system. On other systems, they may indicate a real security problem.

**Sanitizing**

Information, which would be useful to an attacker, has been purposely modified ("sanitized") within this report. This includes the company name, software packages and versions, and system use.

## Area: Access Control

### Area Information

#### Description

For Access Control issues, this assessment looked at user-ids, passwords, and account administration.

#### Discussion

There are a number of issues in this area that are not consistent with the desired security level. These include poor password choices, passwords which are not changed, partially disabled user-ids, and descriptive user-id names.

#### Methods Used

Observation, discussion with the System Administrator, and Tiger

Because of the small number of user-ids on this system, a password-cracking program (such as Crack) was not used.

### Satisfactory Practice: Access Is Restricted To Need To Know

Access is restricted to those with a need to know, in this case being the system administrators, Sybase administrators, and operations staff.

### Finding: Poor Password Choices

#### What

The same password is used for both the root account and the Sybase database Administrator account. This password also falls in the category of a commonly guessed pattern.

#### Ranking

Estimated Risk: 9, Estimated Correction Effort: 2, Suggested Correction Order: 1

#### Discussion

Using the same password on multiple accounts results in all of the accounts being affected if the password for one of the accounts is compromised. This is particularly bad is these are important user-ids.

Using a commonly guessed password pattern makes the system vulnerable to password guessing and password attack programs.

#### Occurrences

The following user-ids exhibit one or more of these problems: root, dbadmin.

#### Correction

Choose better passwords. It also would be wise to create a standard for password choices and make it part of the information security policy and security awareness program.

### Finding: Passwords Not Changed

#### What

The same passwords have been used since the system was put into production. The passwords should be changed periodically.

**Ranking**

Estimated Risk: 8, Estimated Correction Effort: 2, Suggested Correction Order: 2

**Discussion**

Using the same password for a long period of time increases the chance that the password will be accidentally disclosed. It also creates the possibility that a person who previously had a need to know still has the password after their need no longer exists. The exposure to guessing is also increased.

**Occurrences**

This applies to all of the user-ids.

**Correction**

Change the passwords immediately. Then set up a schedule and policy for future password changes. This correction could initially be done along with the implementation of the better password choices.

**Finding: Partially Disabled User-Ids**

**What**

A number of user-ids were disabled by using a flag in the password field, however, they still had either a valid shell and/or cron table entries. Some of the cron entries were only comments.

**Ranking**

Estimated Risk: 2, Estimated Correction Effort: 2, Suggested Correction Order: 12

**Discussion**

Although these accounts could only be compromised by someone who already had compromised the system, the use of these user-ids could help an attacker escape notice. A valid shell would allow any attacker with root access to do an "su - *user-id*" which could help cover the attacker's tracks.

A valid cron entry would allow an attacker to set up a job that would automatically run, even if the attacker were not logged into the system. An empty cron entry, although not in itself dangerous, could be activated by an attacker and not be easily noticed by the system administrator.

**Occurrences**

The following user-ids exhibit this problem: adm, bin, daemon, listen, lp, noaccess, nobody4, sys, and uucp.

**Correction**

- Implement the "noshell" tool for these user-ids.
- Remove all empty cron entries.
- Remove packages SUNWbnuu and SUNWbnur

**Finding: Descriptive User-Ids**

**What**

The application account user-id name dbadmin is descriptive.

**Ranking**

Estimated Risk: 1, Estimated Correction Effort: 4, Suggested Correction Order: 15

**Discussion**

A descriptive user-ID gives a small clue about its use, which could be of use to an attacker.

This is a low risk, but difficult fix.

Changing this user-id will not be difficult, but will be time-consuming because of the number of places it is used and how it is used. If you were willing to accept some problems, using a script which does a mass replace would be a simple way of doing it. Most of the effort would then be in testing or correcting the problems that develop.

**Occurrences**

dbadmin is the only user-id that does this.

**Correction**

Change the dbadmin user-id to some other name.

## Area: Directories and Files

### Area Information

#### Description

The Directories and Files looked at how directories and files are set up on the server. Most of these files fall in the category of systems files (files used by the operating system or related software).

#### Discussion

The primary emphasis was on the permissions assigned. There are a number of file permissions in this area that are not consistent with the desired security level. These include world-writable Directories, and world-readable security/audit directories and files.

#### Methods Used

Observation, COPS, Tiger

### Finding: World-Writable Directories

#### What

A number of directories are world-writable.

#### Ranking

Estimated Risk: 5, Estimated Correction Effort: 4, Suggested Correction Order: 10

#### Discussion

World-writable directories allow any user who can access the system (login) to put files into these directories.

The most innocent use of world-writable directories would be to store files to avoid use charges or to make ownership of the file not so obvious. Likely files might be pornography.

A more serious threat is to place a file or a link in the directory in such as way as to trick a system process into doing something bad.

#### Occurrences

The following directories were found to be world-writable and create an exposure:

/var/spool/uucppublic, /opt/splash/, /var/dt/dtpower/schemes/, /var/dt/tmp/, /var/preserve/, /var/spool/lp/fifos/public/, /var/spool/pkg/, /var/spool/uucppublic/, /var/tmp/

/var/run/rpc_door/ needs further investigation.

#### Correction

- Change the permissions and/or ownership.
- Remove UUCP packages SUNWbnuu and SUNWbnur.

### Finding: World-Readable Security/Audit Directories and Files

#### What

/etc/security, and some its file, are world-readable.

**Ranking**

Estimated Risk: 4, Estimated Correction Effort: 2, Suggested Correction Order: 7

**Discussion**

It is a poor security practice to give out any information about what the system is auditing. This is only a medium risk on this system because there is very little logging used (see System Monitoring below).

**Occurrences**

/etc/security, and some its files

**Correction**

Change permissions on /etc/security and its files.

**Finding: World-Writable Directories Referenced During System Initialization**

**What**

There is a world-writable directory referenced during system initialization

**Ranking**

Estimated Risk: ?, Estimated Correction Effort: 2, Suggested Correction Order: 14

**Discussion**

System initialization is a system process. An attacker could put files into world-writable directory to trick the system into doing something unintended.

**Occurrences**

/var/run/rpc_door (inside /etc/rc0.d/K41rpc) This directory is used for temporary system files which do not need to be saved across boots. In this particular instance the directories and files are related to inter-process communication.

**Correction**

A follow-up with Sun should resolve whether it is necessary or safe for this directory to be world-writable.

**Finding: Ownership Group Undefined**

**What**

The group ownership number for this file or directory is not defined in /etc/group.

**Ranking**

Estimated Risk: 1, Estimated Correction Effort: 1, Suggested Correction Order: 18

**Discussion**

Most of the occurrences are related to Java and are created by the unmodified Solaris installation.

The security exposures created are:

- The System Administrator could accidentally add a valid group with the same number. The members of the new group would have inappropriate access to the files that were found.
- The presence of this problem generates assessment noise that could obscure issues that are more important.

**Occurrences**

Many files, including /usr/java1.2.

**Correction**

Add a group with this group number.

## Area: Physical Environment

### Area Information

#### Description

This part of the assessment looked at things like physical access to the computer, power, and air conditioning.

#### Discussion

Moving the server to the computer room would correct all of the deficiencies.

#### Methods Used

Observation, discussion with the System Administrator

### Finding: System is Not in a Controlled Environment

#### What

The production "dbprod" system is not in a controlled environment.

#### Ranking

Estimated Risk: 6, Estimated Correction Effort: 3, Suggested Correction Order: 3

#### Discussion

The production "dbprod" system is on the System Administrator's desk. This was fine when the system was in development and test phases of the project. For a production system, it is not appropriate. Fortunately, moving the production system into the more-tightly-controlled computer room will address all of the issues.

People Access: The area where the system is located is normally open to anybody with building access. After business hours, the access is controlled, but still open to about 50 technical and non-technical staff including contract janitorial service.

This creates a security exposure because it is relatively easy to defeat the need for a root password by a series of power cycling.

UPS: The system is not on an uninterruptable power supply. By not being on a UPS the system is more vulnerable to power fluctuations and outages.

Dust: The current area is dusty and can cause disk and backup media failures.

#### Occurrences

People access, UPS, dirty environment

#### Correction

Move the production system into the more-tightly-controlled computer room.

## Area: Privileged Environment

### Area Information

#### Description

For Privileged Environment issues, this assessment looked at what privileged users (primarily "root") could do and whether there were weaknesses in the setup.

#### Discussion

The only weakness in this area was one unusual setuid script.

#### Methods Used

COPS, Tiger, observation

### Finding: Privileged Scripts Found

#### What

The script /etc/lp/alerts/printer has the suid (privileged) bit on.

#### Ranking

Estimated Risk: 6, Estimated Correction Effort: 3, Suggested Correction Order: 9

#### Discussion

Privileged scripts are dangerous for a number of reasons:

- It is easy for an attacker to modify a script (compared to a compiled program).
- It is difficult to write a truly secure Unix script.
- Many Unix operating systems have internal exposures that could allow the script to be compromised.

In this case, the suid owner is lp. This makes it much less vulnerable than a root-owned script, but it could still be exploited.

This particular script is a skeleton that can be modified to use for printer fault recovery. This script could be made more secure by doing one of the following:

- Delete it (if it is not used)
- Rewrite the script as a program
- Write a wrapper program which uses exec to run the commands
- Change the permissions (if this will work in the I-T environment).

#### Occurrences

The /etc/lp/alerts/printer is setuid.

#### Correction

Modify or delete.

## Area: System Administration Practices

### Area Information

#### Description

System Administration Practices looked at the way system configurations were made, how they were documented, and how they were communicated. This area also looked at how vendor fixes were applied.

#### Discussion

System Administration practices found weaknesses in Solaris security fixes not being up-to-date. Change control, although not causing any problems yet, is very marginal.

#### Methods Used

Observation, discussion with the System Administrator

### Finding: Security Fixes Not Up-to-Date

#### What

Security Fixes Not Up-to-Date

#### Ranking

Estimated Risk: 8, Estimated Correction Effort: 4, Suggested Correction Order: 6

#### Discussion

No security patches have been installed since the initial Solaris and "dbprod" install. Unfortunately, most attacks exploit known vulnerabilities. It is important that these exposures be minimized.

#### Occurrences

All Solaris security fixes

#### Correction

Install security fixes as soon as they are stable.

### Finding: Marginal Change Control

#### What

Presently the System Administrator does no formal tracking of system changes. Comments, when possible, are put into the changed files. Version control is mimicked in a crude way by copying a file to a name with the date appended before the change is made.

#### Ranking

Estimated Risk: 3, Estimated Correction Effort: 6, Suggested Correction Order: 9

#### Discussion

This approach to changes has a myriad of deficiencies. Some of those are:

- The history of a change is, for the most part, limited to the System Administrator's head. This is a significant exposure if the System Administrator is not available when a problem occurs.
- It is impractical to store a long history of the changes by relying on the date-stamped file name. A tool like sccs or rcs could be used to advantage.

- Some file formats, an example being the password file, do not allow internal comments, so the reason for the change is not documented.
- Management is unaware of most changes being made. This increases the risk of damage from a disgruntled employee.

To the System Administrator's credit, the use of the date copies and internal comments is faithfully performed.

**Occurrences**

Most changes

**Correction**

- Develop a source control/versioning strategy.
- Management should sign-off on all changes.

## Area: System Configuration

### Area Information

#### Description

For system configuration issues, this assessment looked at how this system was configured to accomplish its mission.

#### Discussion

There are a number of services that do not need to be run.

#### Methods Used

Observation, nmap

### Finding: Unneeded Services Available

#### What

Services that are not required on a particular system are not disabled.

#### Ranking

Estimated Risk: 8, Estimated Correction Effort: 2, Suggested Correction Order: 5

#### Discussion

"Dbprod" is used only in a very specific way. Because of this, a number of services can be disabled. Services that are not required on a particular system should be disabled so they minimize the security exposure.

#### Occurrences

The following services currently running can be disabled:

echo, discard, daytime, chargen, ftp, time, finger, sunrpc, exec, uucp, lockd, X11, font-service, sun-answerbook.

#### Correction

- Disable the unneeded services.
- Remove packages SUNWbnuu and SUNWbnur

### Finding: UUCP Installed

#### What

UUCP is installed on this system.

#### Ranking

Estimated Risk: 5, Estimated Correction Effort: 2, Suggested Correction Order: 19

#### Discussion

UUCP is installed with the default Solaris 8 install. UUCP is a complex package and securely configuring it is laborious. UUCP contributes to many of the security weaknesses identified elsewhere. It is recommended that the two Solaris UUCP packages be removed (with pkgrm).

#### Occurrences

The following services currently running can be disabled:

echo, discard, daytime, chargen, ftp, time, finger, sunrpc, exec, uucp, lockd, X11, font-service, sun-answerbook.

**Correction**

Remove packages SUNWbnuu and SUNWbnur

## Area: System Monitoring

### Area Information

#### Description

The assessment of system monitoring looked at what was logged and how the logs were analyzed.

#### Summary

The system logs are only examined occasionally. There also is no intrusion detection.

#### Methods Used

Observation, discussion with the System Administrator

### Finding: No Regular Logging and Exception Reporting

#### What

The system logs are only occasionally examined

#### Ranking

Estimated Risk: 4, Estimated Correction Effort: 3, Suggested Correction Order: 4

#### Discussion

The system logs are only occasionally examined

#### Occurrences

/var/adm/messages, /var/adm/sulog, other logs as they are implemented in the future

#### Correction

The system administrators should monitor the logs on a daily basis. To make this less time consuming and more accurate, a script should be implemented which would do exception reporting on the appropriate logs.

### Finding: No Change/Intrusion Detection Process Used

#### What

There is no pro-active change or intrusion detection being done on the system.

#### Ranking

Estimated Risk: 5, Estimated Correction Effort: 7, Suggested Correction Order: 8

#### Discussion

Even in the best-secured systems, an attacker may be able to find a weakness and exploit it. It' important to know if this has occurred.

Tripwire should be implemented to prevent this. Since this system has been network-connected for quite a period, a baseline done now could be invalid. Since there is a new test system, a baseline could be done on that, and then it could be made the production system. Another alternative would be to bring both systems up to the identical software release levels, and then the baselines could be done on both systems and compared. This has a likelihood of finding many discrepancies that could be time consuming to resolve.

**Occurrences**

N/A

**Correction**

Tripwire should be implemented.

## Area: Application and Third-Party Software

### Area Information

#### Description

For application and third-party issues, this assessment briefly looked at the Sybase database installation.

#### Discussion

The only apparent weakness is that the Sybase security fixes are not up-to-date.

#### Methods Used

Observation, discussion with the Database Administrator

### Finding: Sybase Security Fixes Not Up-to-Date

#### What

The Sybase security fixes are not up-to-date.

#### Ranking

Estimated Risk: 2, Estimated Correction Effort: 2, Suggested Correction Order: 8

#### Discussion

As in the case with Solaris fixes, it is important to keep the fixes up-to-date. However, the risk from Sybase vulnerabilities is not nearly as great as it is for the Solaris vulnerabilities. This is primarily because users must be logged into Sybase before they can do anything. They also are much more constrained since most of them will not have the equivalent of unrestricted command-line access within the Sybase server.

#### Occurrences

All Sybase Fixes

#### Correction

Install Sybase fixes as soon as they are stable.

## Area: Application Data and Processing

### Area Information

#### Description

The Application Data and Processing area briefly looked at the CTA application and data.

#### Discussion

The data in the CTA is primarily in a Sybase database. This protects it to a large extent. The very small number of Unix users also protected the data when it is outside of the database.

#### Methods Used

Observation, discussion with the Database Administrator

### Satisfactory Practice: Data Protection

The data in the CTA database uses the Sybase "raw partition." Since this is not a Unix file system, the data unreadable by any normal Unix command. It would take an attacker a great deal of effort to recover any data in these raw partitions.

The data is stored in a Unix-readable form in cases like backup staging areas. This data is from the database, and not used to update the normal database, so changes to this data would not corrupt the data that is actually in the database. In addition, very few users have Unix access to this system, so this data is reasonably protected.

## Area: Backup and Disaster Preparedness

### Area Information

#### Description

For Backup and Disaster Preparedness, this assessment looked at how backups are created and handled. It also looked at possible threats and what the response would be.

#### Discussion

Although the probability is low, the damage could be significant. Therefore backup and disaster preparedness should be addressed.

#### Methods Used

Observation, discussion with the System Administrator, discussion with the Database Administrator

### Satisfactory Practice: System Recovery

There are a number of recovery methods that could be used in the event that the Sun system running CTA were to have a serious failure. The maintenance contract would be the first line of defense. In addition, the test system has the fortunate situation of having an identical configuration to the production system. The test system will also be physically separated from the production system once the production system is moved into the computer room. For a larger catastrophe, it would be reasonable that an adequate Sun system could be purchased in a relatively short amount of time.

### Finding: Infrequent Backups

#### What

The backups are only done occasionally.

#### Ranking

Estimated Risk: 7, Estimated Correction Effort: 1, Suggested Correction Order: 3

#### Discussion

The backups of the system are done by the System Administrator but only occasionally, usually every other week.

#### Occurrences

N/A

#### Correction

Do scheduled, frequent backups.

### Finding: Backups not stored off-site

#### What

The backups are not stored off-site.

#### Ranking

Estimated Risk: 4, Estimated Correction Effort: 1, Suggested Correction Order: 17

**Discussion**

The backups are all stored on-site. This puts them at risk from inside attacks as well as disaster events.

**Occurrences**

N/A

**Correction**

Store backups off-site. For the volume generated by CTA, an informal method, such as having the Database Administrator take them home, would be adequate.

## Recommendations

The Summary Table (after the Executive Summary) gives a list of the results of this assessment, broken down by area and finding. It includes an estimate of the risk, what corrective action is needed, an estimate of the relative effort to correct the finding, and recommendation of what order of correction would remediate the findings in the best way.

## Appendix 1: References


- AUSCERT, http://www.auscert.org.au/ In particular ftp://ftp.auscert.org.au/pub/auscert/papers/unix_security_checklist
- CERT, UNIX Configuration Guidelines, http://www.cert.org//tech_tips/unix_configuration_guidelines.html
- Garfinkel & Spafford, Practical Unix Security, c. 1991
- SunSoft/Sun Microsystems, Solaris Administering Security, Performance, and Accounting, c. 1993
- Sun Microsystems, Administering Security on Solaris
- Tiger, tigexp utility (see Tiger discussion below) ftp://net.tamu.edu/ftp/security/TAMU/tiger-2.2.4p1.tar.gz

## Appendix 2: SANS Assessment Cross-Reference

| For | See |
|---|---|
| Executive Summary | Executive Summary |
| Operating System Vulnerabilities | All Of The Areas, particularly System Configuration |
| Configuration Vulnerabilities | Area: System Configuration |
| Risks From Installed Third-Party Software | Area: Application And Third-Party Software |
| Administrative Practices | Area: System Administration Practices |
| Security Patches Up-To-Date | Finding: Security Fixes Up-To-Date |
| Sensitive Data Stored Encrypted And How | Area: Application Data And Processing |
| Data Is Sent Over The Internet Encrypted | N/A (System Not Connected To The Internet) |
| Anti-Virus Software Is Updated | N/A (See Scope Of Assessment. Not Considered Since This System Is Not A Server For Windows' Systems) |
| Access Is Restricted To Those With A Need To Know | Area: Access Control |
| Backup Policies, Disaster Preparedness, Etc. | Area: Backup And Disaster Preparedness |
| Other Issues/Vulnerabilities As Appropriate | All of the areas |
| Prioritized List Of Security Vulnerabilities And Issues | Summary Table |
| Prioritized List Of Recommended Fixes | Findings: Correction<br>Priority: Finding Ranking and Summary Table |
| References | References Section<br>Tool Discussions: References |

## Appendix 3: COPS Discussion

### Description

COPS tool is a host-based utility that checks for a number of Unix security exposures. These checks include anonymous FTP configuration and writable directories and files.

### Version Information

The COPS used was the Perl-based COPS, version 1.04. As downloaded, this version of COPS was not configured for Solaris 2.8, so some minor modifications were made.

### References

ftp://coast.cs.purdue.edu/pub/tools/unix/cops/cops.1.04.tar.gz

### Future Use

It is recommended that COPS or Tiger be run on a regular basis. In this environment, somewhere between weekly and monthly should be adequate.

### Options, Configuration Changes, Customization

SECURE_USERS changed from foo@bar.edu to this systems assessment user-id.

### Result Summary

**Message Text:**

Warning! *Directory-Or-File* Is _World_ Readable!

Instances

/etc/security

Interpretation

Files relating to security and auditing should not be world-readable, lest they divulge information useful to attackers.

**Message Text: Warning! *File* Is _World_ Writable!**

Interpretation

Configuration files should not be world-writable.

Instances

/usr/adm/spellhist Although it could be called a configuration file, exposure of it should be innocuous.

**Message:**

Warning! File *Directory-Or-File* (Inside *Initialization-Script*) Is _World_ Writable!

Interpretation: World-writable directories or files used during initialization could be modified to cause a problem during initialization or later.

Instances:

- /var/mail (inside /etc/rc0.d/K36sendmail) These are appropriate permissions for this directory.

- /var/run/rpc_door (inside /etc/rc0.d/K41rpc) This directory is used for temporary system files which do not need to be saved across boots. A follow-up with Sun should resolve whether it is necessary or safe for this directory to be world-writable.
- /dev/tcp (inside /etc/rc0.d/K43inet) These are appropriate permissions since this is a symbolic link.
- /var/spool/uucppublic these are appropriate permissions for this directory, but UUCP should be removed.

## Appendix 4: NMAP Discussion

### Description

NMAP is a utility, which scans a network or systems for services, such as ftp, which are available on a host. For this assessment NMAP was a convenient way of listing the available services, rather than an important part of the analysis.

### Version Information

NMAP 2.53

### References

www.insecure.org/nmap/

### Future Use

NMAP can be run very infrequently. The configuration of which ports/services are open will seldom change.

### Options, Configuration Changes, Customization

The -f option was used to make "dbprod" host was the only one that was scanned.

### Result Summary

#### Available Services

echo, discard, daytime, chargen, ftp, telnet, smtp, time, finger, sunrpc, exec, login, shell, printer, uucp, listen, lockd, X11, dtspc, font-service, sun-answerbook

#### Unneeded Services

Please see the System Configuration section above.

## Appendix 5: Tiger Discussion

### Description

Tiger is a host-based utility that checks for many possible Unix security exposures including anonymous FTP configuration, mail configuration, cron entries, services (ports), unusual files, writable directories, and privileged files (suid and setgid).

### Version Information

Tiger 2.2.4

### References

ftp://net.tamu.edu/ftp/security/TAMU/tiger-2.2.4p1.tar.gz

### Future Use

It is recommended that COPS or Tiger be run on a regular basis. In this environment, somewhere between weekly and monthly should be adequate. The results from this run should be used for the setuid and setgid lists. Tiger Explain (tigexp), a very nice feature which explains the findings. Tigexp will not be as valuable for future runs since most of the output should be already understood.

### Options, Configuration Changes, Customization

Running the CRACK program disabled, check embedded executables only, various root lists changed to include root-like users such as bin.

### Result Summary

**Message Text: --CONFIG-- [Fsys003c] No Setuid List... Listing All Setuid Files**

**And**

**Message Text: --CONFIG-- [fsys003c] No setgid list... listing all setgid files**

Interpretation

Since the baseline files are not available, Tiger listed all of the setuid and setgid files.

Instances

setuid, setgid

**Message Text: --INFO-- [Fsys004i] The Following Setuid Programs Are Non-Standard:**

Interpretation

Non-Standard setuid programs

Instances

/usr/lib/pt_chmod

**Message Text: --WARN-- [Xxxxx] The Following Files Are Unowned:**

Interpretation

Unowned files

Instances

/usr/java1.2/commapi

**Message Text: --WARN-- [Xxxxx] The Following Files Have Undefined Groups**

**Ownership:**

Interpretation

The group ownership number for this file or directory is not defined in /etc/group

Instances

Many files, including /usr/java1.2

**Message Text: --WARN-- [Acc001w] Login ID *User-Id* Is Disabled, But Still Has A Valid Shell**

Interpretation

This user-id was disabled in the password file, usually by supplying an invalid password field, but it still has a valid shell field. User-id disabled, but still has a valid shell

Instances

adm, bin, daemon, listen, lp, noaccess, nobody4, nuucp, sys, and uucp

**Message Text: -WARN-- [Acc005w] Login ID *User-Id* Is Disabled, But Has A 'Cron' File Or Cron Entries.**

Interpretation

User-id disabled, but still has a cron shell

This user-id was disabled in the password file, usually by supplying an invalid password field, but it still has a valid cron (automatically scheduled) job.

Instances

adm, bin, daemon, listen, lp, noaccess, nobody4, nuucp, sys, and uucp

**Root crontab**

Tiger reported many false positives on "cron entry for root does not use full pathname." This was triggered by the entry not starting with a /, but the shell test symbol **[**.

## Appendix 6: Tripwire Discussion

**Description**

Tripwire is a file monitoring/integrity checker for intrusion detection. It has the side effect of also monitoring system changes.

**Version Information**

1.3.1 ASR

**References**

www.tripwiresecurity.com

**Future Use**

It is recommended that Tripwire be run once per week.

**Options, Configuration Changes, Customization**

Tripwire needs to create a baseline database. It should be done before a system is connected to the network. See the Intrusion Detection finding for more details.

**Result Summary**

A baseline database should be created and updated as needed. Any anomalies that are found in future checks should be investigated.