Oracle® Solaris Studio 12.4: Security Guide

Part No: E37092
October 2014

ORACLE

Contents

Using This Documentation

- **Overview** – Describes the security issues users need to be aware of with this Oracle Solaris Studio 12.4 release.
- **Audience** – Application developers, system developers, architects, support engineers
- **Required knowledge** – Programming experience, software development testing, aptitude to build and compile software products

Product Documentation Library

Documentation, late-breaking information, and known issues for this product are included in the documentation library at `http://docs.oracle.com/cd/E37069_01`.

The known issues for this Oracle Solaris Studio release can be found in the chapter "Known Problems, Limitations, and Workarounds in This Release" in "Oracle Solaris Studio 12.4: Release Notes ". For more information about compilers and tools, see the relevant man pages or Help.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Feedback

Provide feedback about this documentation at `http://www.oracle.com/goto/docfeedback`.

♦ ♦ ♦ CHAPTER 1

Oracle Solaris Studio Security Information

This document includes information about the following:

Oracle Solaris Studio Security Considerations

Oracle Solaris Studio is a suite of compilers, debuggers, and analysis tools, and an integrated development environment (IDE), for developing, debugging, and tuning applications for the Solaris and Linux platforms. Like other development tools, the Oracle Solaris Studio compilers and tools are intended to be used in an environment that is isolated from production environments since these tools can accessed by users to manipulate applications during execution. While production environments are typically the focus of security considerations, developer tools and development environments should also be considered from a security perspective.

System administrators play an important role in determining which assets require protection and putting in place controls and policies to protect these assets. By itself, Oracle Solaris Studio does not provide any access to assets or operating environment features that the user does not already have. The risk that Oracle Solaris Studio adds is that it allows users who have gained un-entitled access to assets or systems by means that don't include Oracle Solaris Studio to use the capabilities of the Oracle Solaris Studio developer tools to cause a security breach. A user who is running the Oracle Solaris Studio tools has access to all of the capabilities exploited by the debuggers and analyzers by using the operating system interfaces directly. But the Oracle Solaris Studio tools make it easier to understand and use these operating system capabilities to probe the internals of applications, manipulate hardware registers, memory, and stack, and control the execution of an application.

Precautions for System Administrators and Users Installing Oracle Solaris Studio

Oracle Solaris Studio compilers and tools are intended primarily for use in development environments. If Oracle Solaris Studio is needed in a production environment (for example, to debug a production application or analyze a performance bottleneck), take measures to limit access to these tools. Install only the Oracle Solaris Studio components needed for development or production tasks. The Oracle Solaris Studio package installer lets you select which Studio components to install.

Oracle Solaris Studio IDE lets you install non-Oracle supported plugins. Before downloading any third-party software such as these plugins, assess the security safety of such plugins.

Keep installations of Oracle Solaris Studio current with the latest patches, especially security patches.

Oracle Solaris Studio Performance Analyzer requires elevated privileges for certain debugging and analysis tasks. Provide these privileges through temporary accounts, and monitor these accounts accordingly.

Precautions for Developers

Oracle Solaris Studio compilers and tools create output files such as logs, core dumps, and object files. The permissions on these files are set using the user's default permissions. To protect the output files from unwanted access, limit the default permissions to allow only access that is absolutely needed. Users set the default permissions using the Solaris and Linux `umask` command.

Using Oracle Solaris Studio Libraries

Oracle Solaris Studio includes a set of libraries that provide runtime support on supported platforms: performance libraries targeted for compute-intensive applications, and debugging and performance analysis libraries used in tuning applications in a development environment. Performance and runtime libraries are used in production environments and are installed by system administrators as required for the applications that will be run.

As the name implies, performance libraries are optimized for performance, which means that data checking is kept to a minimum insuring maximum performance. When using performance libraries, the application developer is responsible for validating data being passed to these libraries.

Using Remote Development in the IDE

Using remote build hosts in the IDE requires login credentials. Compromised security on the client system running the IDE might lead to unauthorized access of remote server hosts. In shared desktop environments, storing login credentials is not advisable for situations requiring a high level of security.

Another area of consideration with respect to remote development is the caching of source code on client systems during remote development. To increase IDE performance and responsiveness, the remote development feature in the IDE caches files from the server, including source code, on the client machine. The cache folder on the client machine is *user_directory*`/var/cache/remote-files`.

On Solaris and Linux platforms, *user_directory* is: `~/.solstudio/ide-`*version*`-`*OS*`-`*architecture* (for example, `~/.solstudio/ide-12.4-SunOS-i386`).

On Microsoft Windows platforms, *user_directory* is `~/Application Data/.solstudio/dd-`*version*.

On Mac OS X platforms, *user_directory* is `~/Library/Application/Support/solstudio/dd-`*version*.

In sensitive security environments, take care with this cache folder, including deletion or encryption.

Using Remote Development in Performance Analyzer

The Performance Analyzer remote client requires a user name and password to connect to the remote server. The user name is stored to the disk in the user's home directory, but the password is never stored to the disk.

The Performance Analyzer remote client uses a Java implementation of `ssh` (`jsch` version 0.1.49) for connection to the `sshd` server on the remote host.

The Performance Analyzer client stores information in the file `~/.solstudio/analyzer-12.4/analyzer.xml` on all client platforms including Oracle Solaris, Linux, MacOS, and Microsoft Windows.