The Safe-Tcl Security Model

John K. Ousterhout
Jacob Y. Levy
Brent B. Welch
Sun Microsystems Laboratories
2550 Garcia Avenue, MS UMTV-29-232
Mountain View, CA 94043-1100

Presenter: Guoyu Deng

Outline

- Introduction
- TCL overview
- Security model
- Conclusion

Introduction

When you download an executable, you don't know who wrote it, and you can't trust it.

Safe-Tcl provides a model which provides safety when you are running a program written by others.

Safe-Tcl defends against attacks on integrity and privacy. (Defense is not so necessary against Denial-of-Service attacks.)

TCL Overview

- **"Tcl (tool command language, pronounced 'tickle') is the industry's first scripting language capable of handling enterprise-scale integration tasks. It's used by over half a million developers worldwide and has become a critical component in thousands of corporations ..."**
  http://www.people-ef.net/eggdrop/tcl.htm
- **Tcl is an interpreted scripting language, similar as Unix shell programs.**

TCL Overview (2)

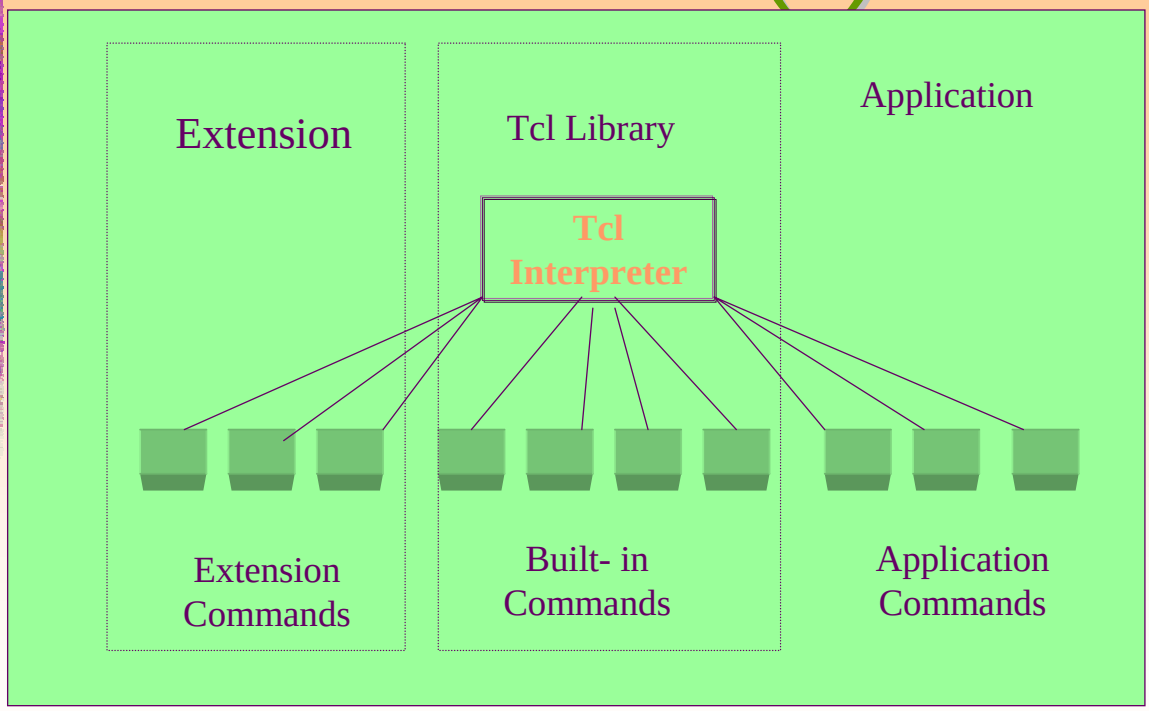


Figure 1. Shows Tcl is embeddable and extensible

Four Properties of TCL

1. Tcl is an interpreted language. There is a natural place to add security controls.
2. Tcl is safe with respect to memory usage: it has no pointers, array references are bounds-checked, and storage is managed by the interpreter.
3. A single Tcl application can have more than one interpreter. The interpreters are totally disjoint from each other.
4. Different interpreters can have different command sets with different security properties.

Goals of Safe-TCL

- **Safe-Tcl is a security mechanism for controlling the execution of Tcl scripts to prevent attacks to the computer system.**
- **It mainly protects system integrity and privacy.**

Security Model

An “applet” is an untrusted program or script.

An “application” is a trusted environment (including the interpreter) in which an applet may run.

An applet calls the application to execute “commands”.

Unsafe Commands

Commands	Functionality
open,socket	Open files / internet connections
file,glob	Files management
exec	Invoke subprocesses
load	Load shared libray binary into app. From file

Master and Safe Interpreter

The “master interpreter” is a fully functional interpreter;

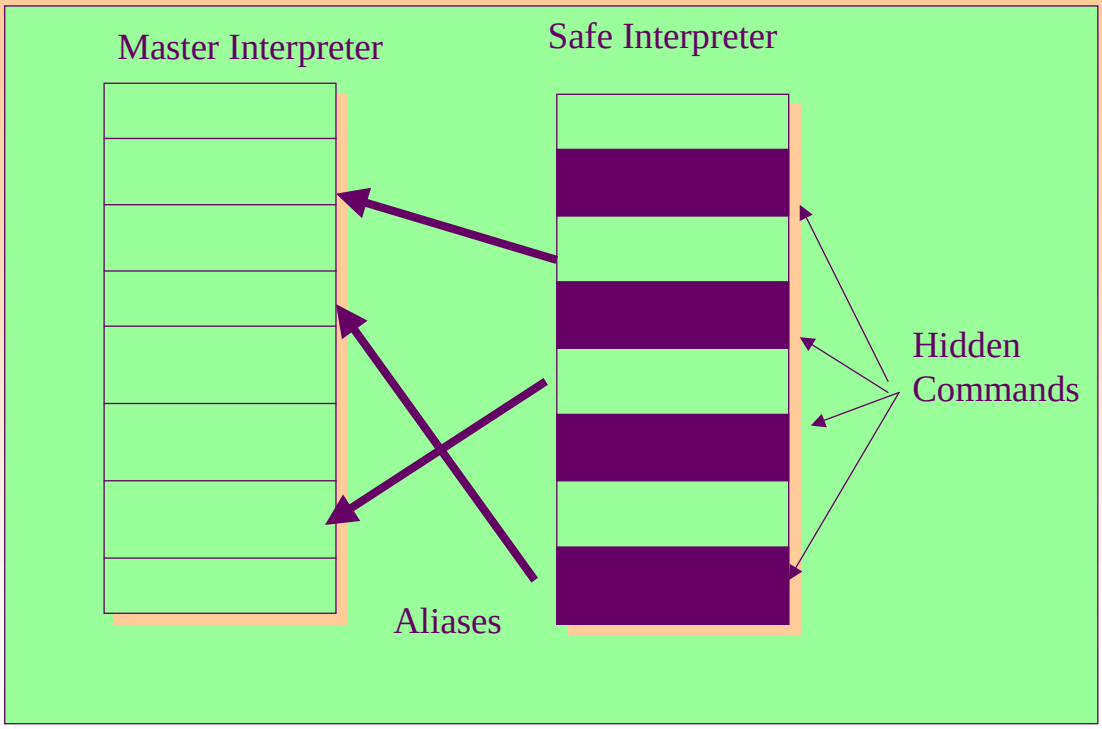
A “safe interpreter” is an interpreter with limited functions;

An “alias” is a command call from a safe interpreter.

A “hidden command” is

A “padded cell” is a safe environment for running applets: it has a safe interpreter, aliases for running commands, and it can’t run hidden commands.

Padded Cell



Conclusion

Safe-Tcl security model has some strengths:

- It separates untrusted code from trusted code with clear and simple boundaries between environments having different security properties.
- It groups data and code with similar security properties together. It do no prescribe any particular security policy, which reduced amount of code used.
- it only provides mechanisms for implementing a variety of security policies. Different organization can implement different policies depending on their needs.

Although long article but easy to follow and understand.

Fast reference (most referred articles are available on web sites.)

Questions

1. Which is better comparing with Java's model with Safe-Tcl?

2. To deal with Denial-of-Service Attack to Safe-Tcl, the author suggests to kill the applet or restart the system, it is a good idea?