System Administration
Final Project

April 24, 2006

1 Project Description

The final project will consist of two parts: setup of a LAMP server and simulation of a real-world server. In the first part, students will install all the components of LAMP and secure the operating system and daemons. The second part will be divided into two activities: creation of four untrusted user accounts on the system and attempts at being malicious users on another system.

2 Project Specification

2.1 Setup of a LAMP Server

The following services and programs should be installed and working on the server:

- SSH on port 1`XX`22
- Apache with suExec on port 1`XX`80
- PHP running as CGI under suExec
- MySQL

where `XX` is your group number. You should also install the following programs on the server

- `nmap` - a port scanning program. This program should only be accessible to administrators.
- `links` - a text-mode web browser.
- `mutt` - a text-mode mail client.

Quotas and access control lists should be working, and the tools for working with access control lists should be available to users on the system.

2.2 Server Simulation

2.2.1 Preparing Your Server

You need to create four user accounts: two with complete shell access and two that can only connect via `scp` and `sftp`. Each account should also be provided with complete access to a MySQL database. Please mail the login information and MySQL information for these accounts to the `dima+decal@ocf.berkeley.edu`.

You should assume that all user accounts will be used by malicious users. In other words, you should assume that a user account poses a threat to your server and other users. Therefore, you should set appropriate restrictions on each account. However, the accounts should be able to host their own website and execute PHP scripts that make use of their MySQL database.

You will be responsible for auditing user accounts and ensuring that they have not set insecure permissions on their files. Scripts that check for insecure permissions are highly recommended. You should set the scripts to execute on a regular basis, and have the output of these scripts sent to your email addresses. You may even elect to have the scripts automatically adjust file permissions.

If users manage to break their account in some way, it will be your responsibility to help them fix their account, within the guidelines of the project.

2.2.2 Testing Another Server

You will be provided with four user accounts on another group's server. One of these accounts will have complete access to the system, and the other account will only have `scp` and `sftp` access to the system. Your task will be to act as malicious users and test the security of the server. You should attempt to gain unauthorized access to files and directories, exceed any quotas placed upon your accounts, execute restricted commands, and any other attack within the project guidelines.

You will also be testing the server to ensure that you can use your web space to run PHP scripts that interact with your MySQL database. See the list of suggested scripts to setup below. Please note that if you are using a script of your own choosing, the administrators of the server will not be responsible for any security holes that develop as a result of the script, beyond those involved with file permissions.

3 Project Guidelines

- All submission text should be in your own words. For example, do not just copy and paste the explanation for a command from its man page, and do not just copy and paste the package description from `apt-cache` when describing a package.
- Attacks against remote systems, other than the servers to which you have been granted access, should not be used and are in violation of campus network policy.

- Attacks that consume all the processing power of a server should not be used and will not be considered valid.

4 Project Hints and Help

- If you need help or have any questions, please send an email with your group number to `sysadmin-decal@ocf.berkeley.edu`.
- When choosing a package to install, it is usually best to choose the package with the simplest name. For example, `apt-cache` presents you with the choice between installing `foo-server` and `foo-server10`, `foo-server` will probably be the best package to install.
- If you wish to completely remove a package, recall that you may pass the `--purge` parameter to apt-get. For example, to completely remove package `bar`,

4.1 Suggested Internet Scripts

The following are PHP scripts that make use of a MySQL database. Please note that the installation documentation for some of these scripts specifies insecure actions that you should not perform blindly; using what you know, figure out a more secure method.

- gallery: gallery.sf.net
- wordpress: www.wordpress.org
- drupal: www.drupal.org

5 Submission Guidelines

Please submit a listing of all important commands you used to setup your server, an short explanation for each command, and whether root access was needed to execute the command. If necessary, provide the context in which you used the command (ex., the directory you were in when you executed the command) or the output of a command. With regards to package installation, you should also provide a description of why it was necessary to install the package and, if applicable, why you chose it instead of other similar packages.

6 Project Grading

The grading system for the project will be finalized shortly. For now, assume that both parts of the project are equally weighted.