Diesel: Applying Privilege Separation to Database Access

Adrienne Porter Felt, **Matthew Finifter**,
Joel Weinberger, and David Wagner
{apf, finifter, jww, daw}@cs.berkeley.edu

UC Berkeley

March 23, 2011

Introduction

- Applications often give complete database access to all program modules
- SSH daemon uses privilege separation to protect its private key (Provos et al., Usenix Security, 2003)
- *Data separation*: a design pattern in which each program module receives access to only the data it needs
- Privilege separation provides defense in depth for specific resources against program flaws
- Data separation does the same for database data

Talk outline

- Introduction and problem
- Benefits and use cases for data separation
- Design
- Implementation
- Experience

Benefits of data separation

- Additional line of defense against software defects
  - Vulnerabilities
  - Logic flaws
- Simpler code review
  - Can prioritize modules with access to critical data

Use cases for data separation

- Capability-secure programming
    - Data subsets are capabilities
- Web applications
    - Can reduce privileges by user, module, or both
- Secure extensibility
    - Third-party module gets only the data it needs

Terminology

- *Restricted connection*: a database connection limited by a policy to a subset of data.
- *Data separation framework*: a library for setting and enforcing policies for restricted connections.

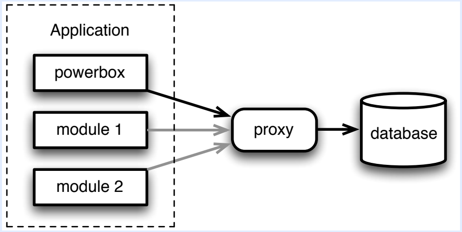
Using restricted connections

- One powerful program module (the powerbox) has full database access.
- This module creates restricted connections with policies.
- It distributes them to other modules.
- A module receiving a restricted connection can treat it like a regular connection.

Paring down a connection

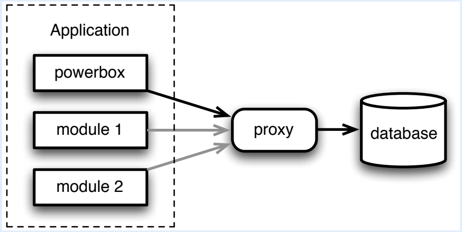
- Layering policies on top of one another
- Allows for fine-grained data separation
- Enables incremental deployability

Diesel Architecture (1)



- Does not require DBMS support or awareness
- Language-neutral proxy-based architecture
- Proxy applies and enforces policies

Diesel Architecture (2)



- All modules talk to the proxy, which talks to the database
- Powerbox has unrestricted database connection
- Other modules have restricted connections

Policies

- Policies are sent from the application to the proxy
- Policies consist of SELECT, INSERT, UPDATE, and/or DELETE privileges for tables or views
- Once set, connection's policy can never be made less strict

Proxy

- Plugin for MySQL Proxy
- The proxy interprets policy-setting commands and maintains state for each connection to it
- Other commands are passed through to the database if the policy allows
- Proxy checks commands against policy by parsing commands for operation (e.g., SELECT) and table names

Experience

- JForum
  - Forum web application written in Java
  - Data-separated each module
- Drupal
  - Content management system written in PHP
  - Data-separated third-party Brilliant Gallery extension
  - Data separation makes vulnerability severity negligible
- WordPress
  - Blog web application written in PHP
  - Data-separated third-party WP-Gallery extension

Conclusion

- Introduced *data separation*
- Implemented prototype data separation framework called Diesel
- Refactored 3 applications to use Diesel
- More details in the paper!

Thank you!

Matthew Finifter, `finifter@cs.berkeley.edu`