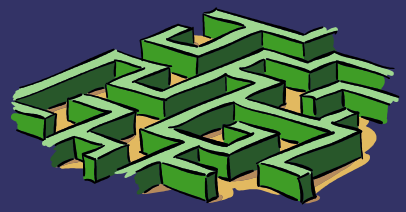
MySQL Security

Boston MySQL User Group
Monday, February 8th, 2007

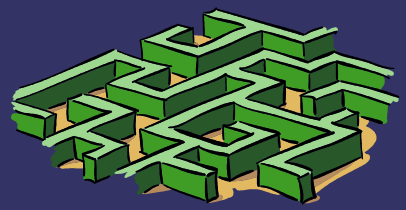
Sheeri Kritzer, MySQL DBA
http://www.sheeri.com
awfief@gmail.com

Technocation, Inc.
http://www.technocation.org



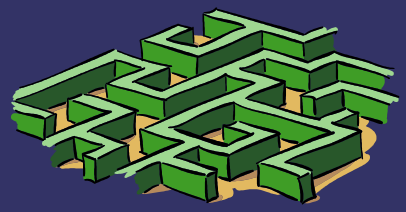
Overview

- ACLs
- Test dbs & anonymous accounts
- OS files and permissions



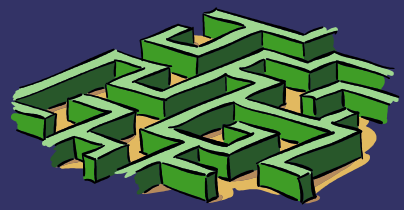
Overview

- Application data flow
- SQL Injection
- XSS



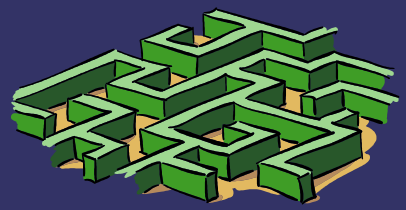
ACLs - Who Has Access?

- SELECT user,host,password FROM mysql.user;
- SUPER
- Anonymous



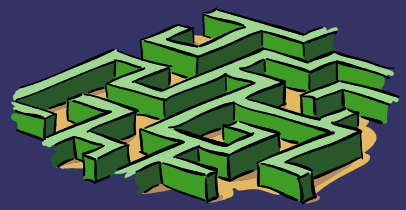
ACLs – From where?

- %
- %.company.com
- 192.168.% or 10.0.%



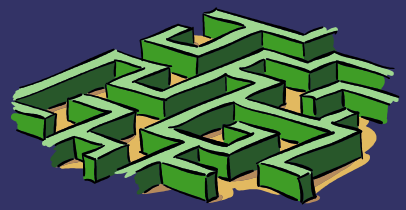
ACLs – From where?

- localhost, --skip-networking
- firewall
- DOS



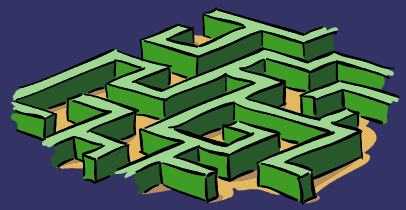
ACLs – To Do What?

- --local-infile=0
- --skip-symbolic-links
- GRANT
  - MAX_QUERIES_PER_HOUR
  - MAX_UPDATES_PER_HOUR
  - MAX_CONNECTIONS_PER_HOUR



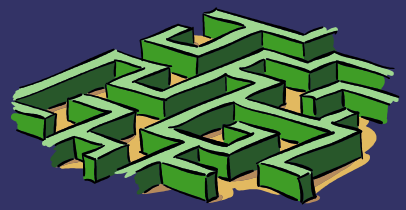
Server Options

- --bind-address
- --skip-name-resolve
- --skip-show-database



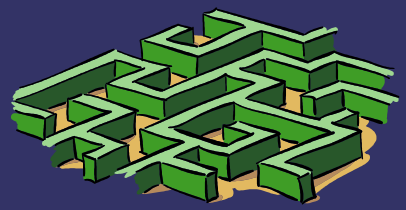
Changing ACLs

- How are ACL changes audited?
- When do ACL changes happen?



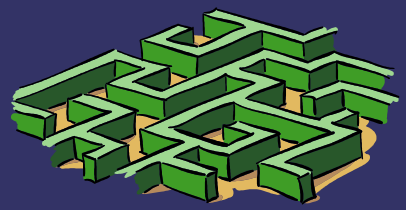
Test Databases

- Why get rid of them?
- Copying tables
- Stuff with data



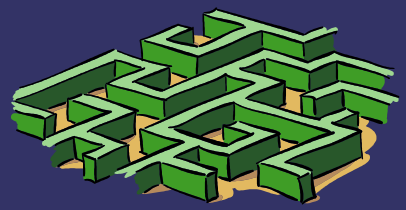
OS Files and Permissions

- mysql server user
- mysql server files & logs
- Passwords on commandline
- Office policies/runbook



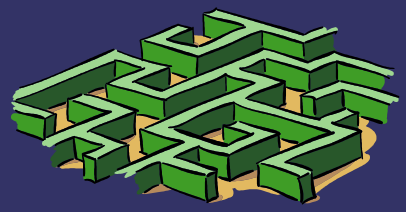
OS Files and Permissions

- Backups
- /etc/my.cnf, my.ini, .my.cnf
- CLI, GUI tools
- Personal history files



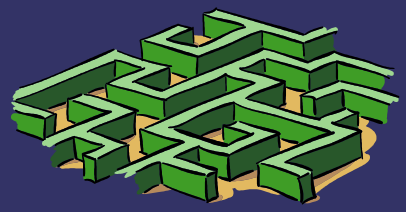
How Does Your Data Flow?

- Where is user data encrypted?
- Where do errors go?
- Where does the traffic flow?



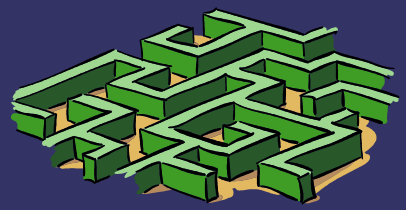
Administrative Applications

- Same data, different interface
- Reporting
- VPN
- “It's public” vs. “It's easily accessible”



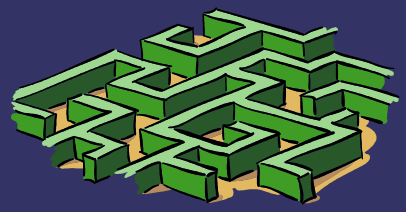
Plaintext Passwords Are Bad!

- Storage of customer login
- Compromised DB
- Transmission of passwords/hashes
- Users may use elsewhere



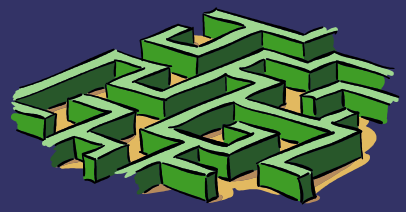
Plaintext Passwords Are Bad!

- Where are you encrypting?
- Where are you checking?



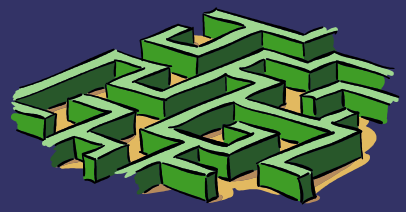
Validate User Input

- ; \g \G ' “ UNION
- HTML encoding
- NULL / char(0)
- VARCHAR and ' '



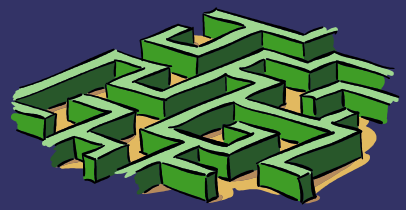
Validate User Input

- Save yourself time
- Buffer overflows
- CHARSET



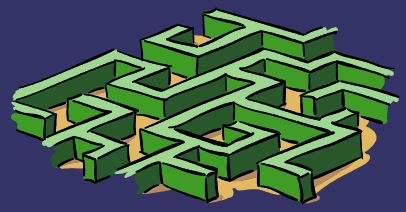
Trusting GET or POST

- Only from certain pages
- Even with valid session ids, cookies
- register_globals=off in PHP



Use Prepared Statements

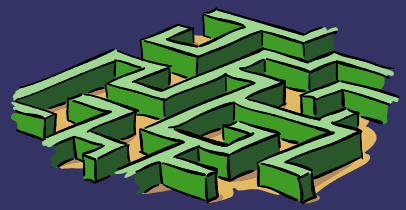
- Code
  - SELECT fld1,fld2 FROM tbl1 WHERE fld1=?
  - $prep=prepare($query)
  - execute($prep,@variables)
- Prepared statement speed
- Stored procedures



Encryption

- SSL is per-client
- Unencrypted MySQL data streams

```
shell> tcpdump -l -i eth0 -w -src or dst port 3306 | strings
```



Feedback?

- Other ideas....