Why Are Databases So Hard To Secure?

Sheeri Kritzer Cabral
Database Administrator
The Pythian Group, www.pythian.com

cabral@pythian.com
Shmoocon 2008

THINK

- I want your balls!
- Ask for information
- Respect me, respect you

My Qualifications

- Guardium, www.guardium.com
- SA starting May 2001
- DBA starting Mar 2004

Caveat

- MySQL DBA
- Experience and familiarity with Oracle, SQL Server, Postgres, Sybase, DB2 but not an expert

Pythian's Qualifications

- SOX
- HIPAA
- FDA (ORA/OE)

Pythian's Qualifications

- Credit Cards
  - PCI
  - PCI Gateway

- EPD (European Privacy Directive)

Pythian's Customers

- Western Union
- Palm Coast Data
- www.pythian.com/aboutUs/customers.html

But enough of this palaver!

Let's get this show on the road.....

General Security

- Patch me if you can!
- Prevent Access
- Prevent meaningful knowledge
  - Encryption
  - Permissions

Securing a Network

- Physical Access/Isolation
- Authentication
- Traffic Shaping
  - Content
  - Volume

Securing an Operating System

- Authentication
- Firewall
- Installed programs
- User ACLs

Securing 3rd Party Applications

- Authentication
- Configuration
- Content shaping

Securing Your Applications

- Authentication
- Configuration
- Be wary of user-entered data
  - Even if it's checked elsewhere

What is a Database?

- Structured collection of records

- Database usually means DBMS
  - Storage
  - Retrieval
  - Processing

What is a Database?

- Structured collection of records

- Database usually means DBMS
  - Storage
  - Retrieval
  - Processing

Database Security In General

- Who is responsible?
- Designed to store information
  - even/especially sensitive information
  - varied information
  - related information

Database Security In General

- Information gained in one part can damage another

- Access points can be many
  - DBMS controls permissions
  - OS/Network/Apps control access
  - Applications control interfaces

Application Vulnerabilities

- Compromised interface
  - any data the interface can access might be compromised
  - encryption algorithms can be compromised

But how?

(I'm glad you asked...otherwise I have to stand up here a long time while you throw shmooballs at me)

Auditing and Monitoring

- Prevention is one part of security
- Auditing - review and assess security
- Monitoring – alerting of security issues

Access Points

- Who can login?
  - Network, seeing traffic
    - http://forge.mysql.com/snippets/view.php?id=15
  - OS
    - Data
    - Logs
    - Backups

Who Can Login?

- Database
  - From where? What can they do?
  - What consequences can they bring about?
- Data
  - ACLs on tables, columns, stored procedures
  - Can use VIEWs
  - What is sensitive?

GRANTing Access - Oracle

```
CREATE USER username

IDENTIFIED { BY password | EXTERNALLY | GLOBALLY AS
    'external name' }

[ DEFAULT TABLESPACE tablespace ]

[ TEMPORARY TABLESPACE { tablespace |
    tablespace_group_name } ]

[ QUOTA { integer { K | M }  ON tablespace } | UNLIMITED ]

[ PROFILE profile ]

[ PASSWORD EXPIRE ]

[ ACCOUNT { LOCK | UNLOCK } ]
```

GRANTing Access - MySQL

```
GRANT priv_type [(column_list)] [, priv_type [(column_list)]] ...
    ON [object_type]
        {tbl_name | * | *.* | db_name.* | db_name.routine_name}
    TO user [IDENTIFIED BY [PASSWORD] 'password']
    [REQUIRE        NONE |      [{SSL| X509}]
   [CIPHER 'cipher' [AND]]       [ISSUER 'issuer' [AND]]
        [SUBJECT 'subject']]   [WITH with_option [with_option] ...]
```

GRANTing Access - Postgres

CREATE USER name [[WITH] option [...]]

where option can be:

SYSID uid

| CREATEDB | NOCREATEDB

| CREATEUSER | NOCREATEUSER

| IN GROUP groupname [, ...]

| [ENCRYPTED | UNENCRYPTED] PASSWORD 'password'

| VALID UNTIL 'abstime'

GRANTing Access – Microsoft SQL Server

```
CREATE USER user_name
    [ { { FOR | FROM }
      {
        LOGIN login_name  | CERTIFICATE cert_name
        | ASYMMETRIC KEY asym_key_name
      }
      | WITHOUT LOGIN
    ]
    [ WITH DEFAULT_SCHEMA = schema_name ]
```

Other ACL's

- Object access
- Password policies
- Roles

Who + Where?

- user@host
- Server firewall
- Network firewall

And that's just ACL's!

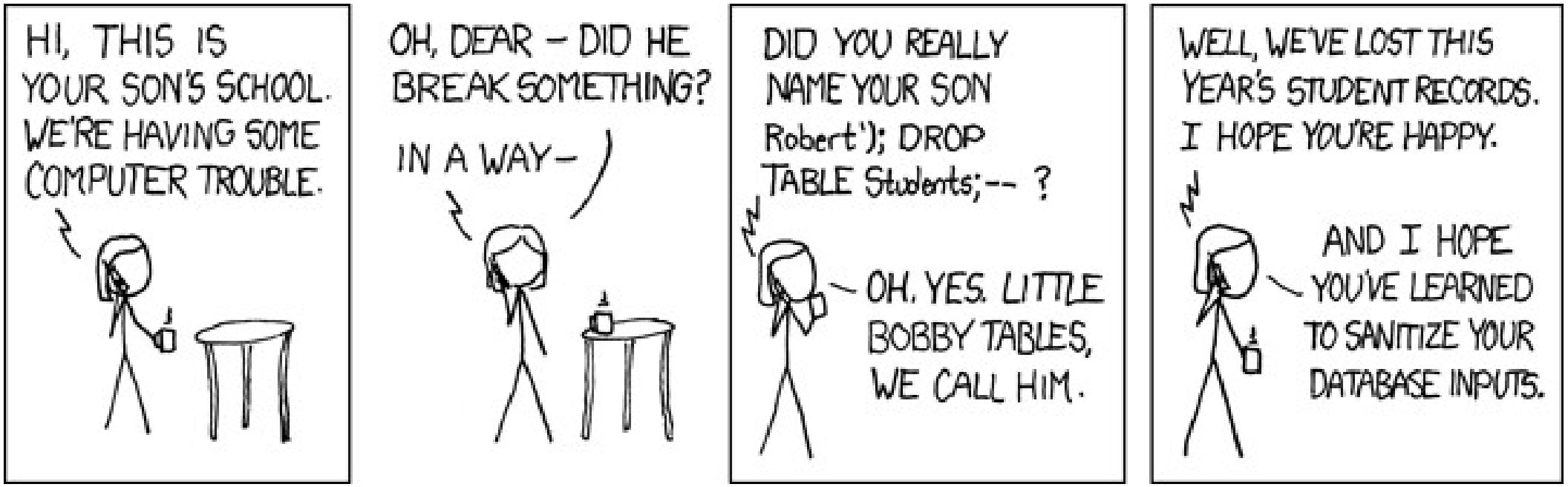
The PYTHIAN GROUP™

your database maestros

Speaking of Users

- Who owns the application user account?
- Who is responsible for that security?

What + How



The PYTHIAN GROUP™

your database maestros

What + How

- Direct access
- Stored Procedures

Encryption

- DB = another phase
- Still a hard issue

Application Insecurities

- Code itself
- Connection information
- Numeric/guessable ids
  - index.php?id=743374

No Sex is Safe Sex

- “Safer”
- Risk assessment/management
- Orgy!

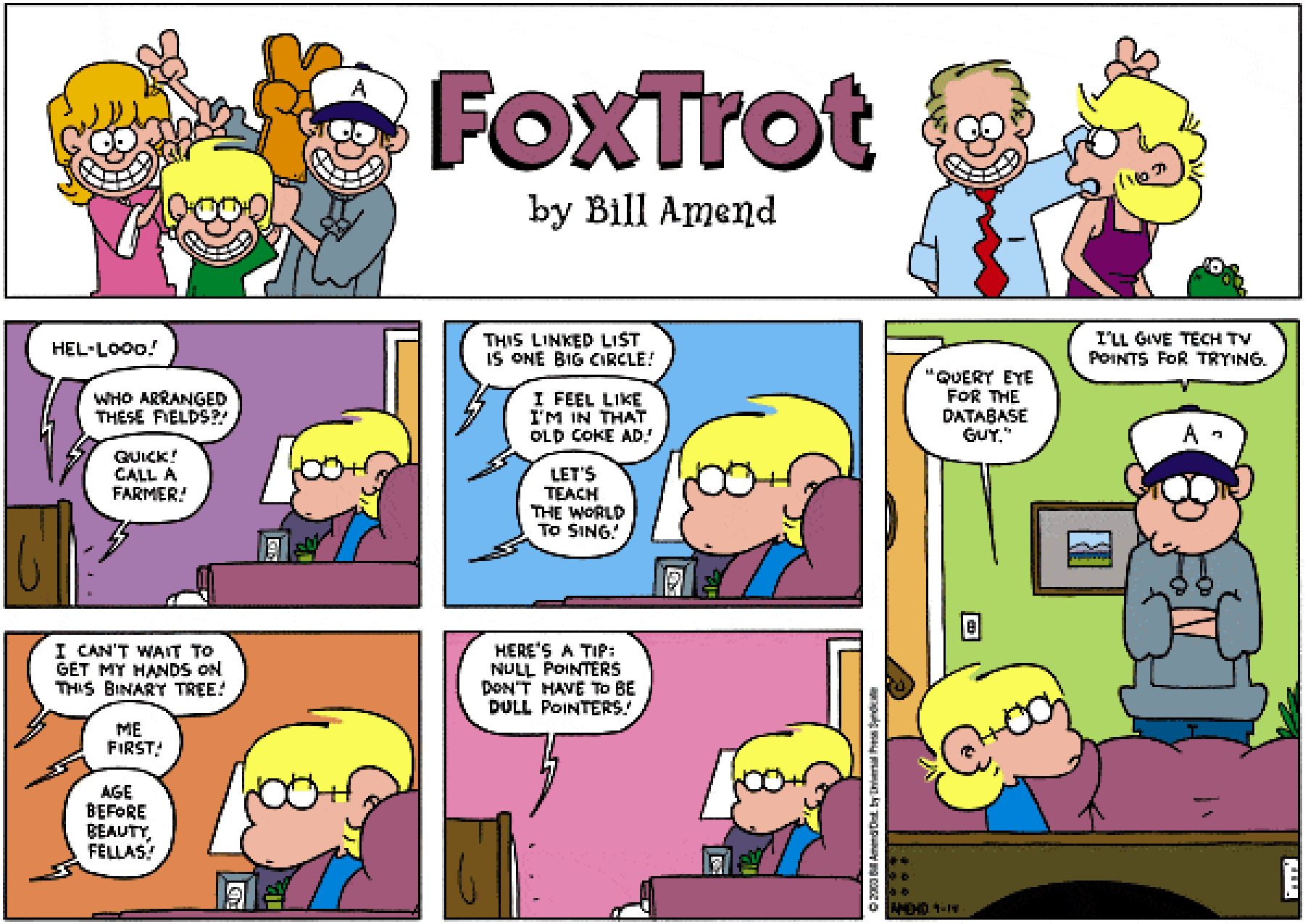
Liars

- Very little you can do

Good DB Security Books

- Implementing Database Security and Auditing, Ron Ben Natan

- Database Security and Auditing: Protecting Data Integrity and Accessibility – Hassan A. Afyouni



Easy to remember:

cabral@pythian.com

www.sheeri.com