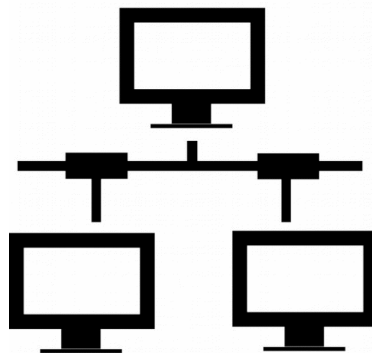
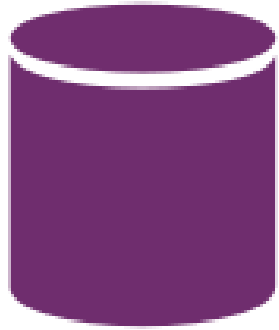


# Toolkit

Python-based security oriented demonstration



# Features

- SQL injection
- Eavesdropping/packet sniffing
- Image steganography

# Application requirements

- **GUI**

- Launcher
  - PyQt
- Steganography (frontend)
  - PyQt

- **Non-GUI**

- Eavesdropping
  - Python
  - wireshark
- Steganography (backend)
  - Python
  - Python Image Library (PIL)

- **Web & database**

- SQL injection
  - CherryPy server
  - SQLite3 database

# SQLi

- Simple CherryPy server to demonstrate how SQL injections are preformed
- Database generated at runtime to limit SQL command reproussions – drop all the tables you like!

## USERNAME LOOKUP

Username lookup by first name (try john)

Show help

[Shutdown Server](#)

Search

Search results:

**SELECT username FROM users WHERE firstname='john'**

jodo3849

**SELECT firstname FROM users**

admin
david
allen
steven
raquel
juliette
john
becky

--'

No results found

Clear results

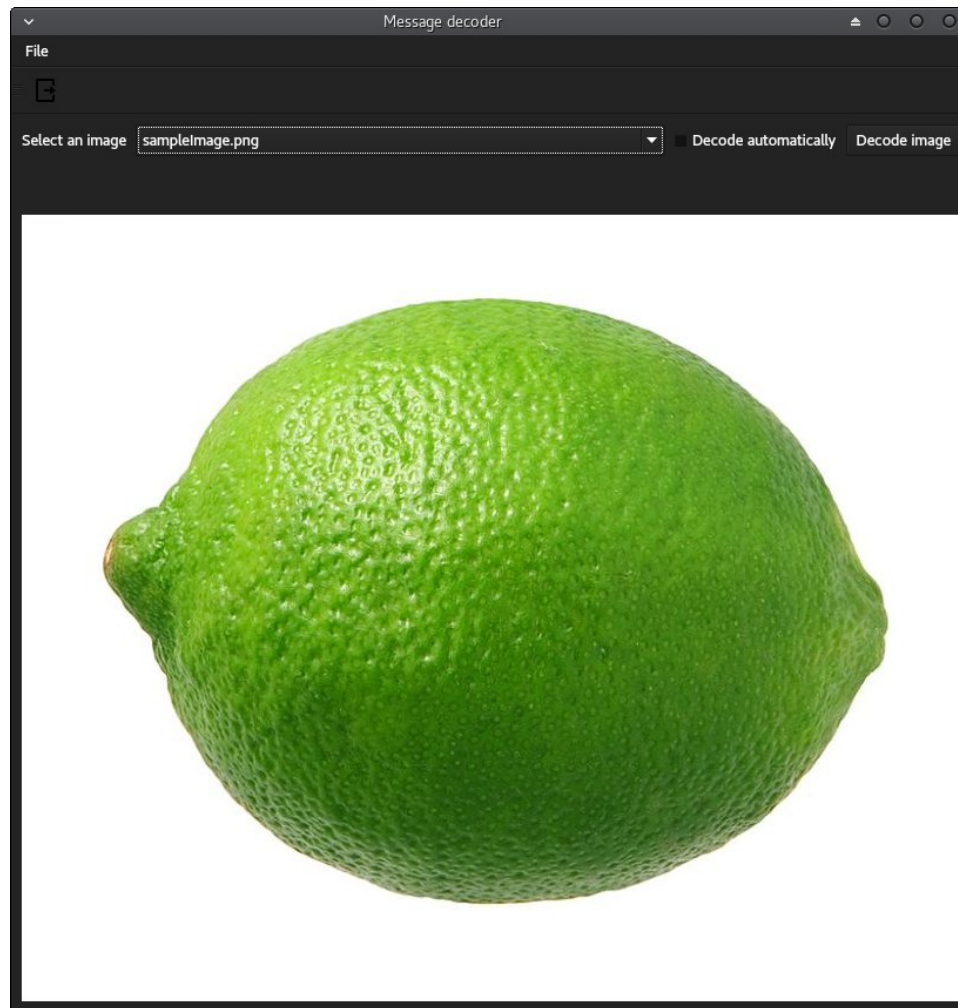
# Eavesdropping

- Sender-receiver relationship for simplicity
  - Supports basic encryption
- Sniffer displays packet data

```
===== Packet Sniffer =====  
[1] Begin capturing packet data (15 seconds)  
[2] Specify capture length  
[3] Display sniffer information  
[4] Quit  
1  
  
Initiating Wireshark  
Capturing on 'Loopback'  
11 packets captured  
[SNIFFED DATA] UCRYPT  
[SNIFFED DATA] UACC  
[SNIFFED DATA] Test message  
[SNIFFED DATA] Test
```

# Steganography

- Encodes a plaintext message into .png image
- Decodes and displays messages



# Unfinished business

## Additional features:

- RSA & Diffie-Hellman for Eavesdropping demo

## Bugs

- Steganography window improperly resizes
- Enter key doesn't execute query on SQL webpage

# Live demo

