10+ Security Tips

by Christian Schmitz

10 Security Tips

- Things you need to remember
- Do and don‘t
- from practice

1. Hashes

- Avoid storing passwords at all
- Avoid storing passwords in clear text or encrypted
- Always hash passwords!
- MD5 or better SHA1, best SHA-512

2. Salt your hashes

- always salt your hashes
- MD5(„Password“) bad
- MD5(„myapp.Password“) better
- MD5(„myapp.Username.Password“) best
- Google for 5d41402abc4b2a76b9719d911017c592

3. Avoid SQL Injection

- Code:
  db.SQLSelect("delete from Users where id = '" + value + "'")
- User enters or hacks for ID: ' or 'h'='h
- you execute:
  delete from Users where City='' or 'h'='h'
- Escape all data.

3. Avoid SQL Injection

- Solution: PeparedStatement

```
// your database
dim db as REALSQLDatabase
// prepare statement
dim p as REALSQLPreparedStatement
p = db.Prepare("delete from Users where id = ?")
// bind data type and value
p.BindType(0, REALSQLPreparedStatement.SQLITE_TEXT)
p.Bind(0, value)
// run statement
p.SQLExecute
```

4. Validate Input

- Don‘t trust user input.
- Is it a number? Number in range?
- Text okay? Not too long? Limit text length.
- Check for invalid characters.
  No „:“ in filenames on Mac or „\“ on Windows.
- Is action now allowed to run? (hacked button)

5. Handle Exceptions

- Don‘t let app crash because of exception.
- app.UnhandledException
- try & catch & finally
- Check documentation what methods throw exceptions. Like picture constructor or binarystream

5. Handle Exceptions

- no code like this:

  volume(0).child(„Library“).child(„Preferences“).Child(„mypref.txt“)

- no code like this:

  listbox1.addrow folder.child(index).name

- Check for nil and handle exceptions!

5. Handle Exceptions

- What is wrong here?
- for i as integer = 1 to folder.count
  if folder.child(i).name = „test.txt“ then
    folder.child(i).delete
  end if
next

5. Handle Exceptions

- What is wrong here?
- for i as integer = folder.count downto 1
  dim item as folderitem = folder.trueitem(i)
  if item<>nil and item.name = „test.txt“ then
    item.delete
  end if
next

5. Handle Exceptions

- dim files() as folderItem
  dim c as integer = folder.count
  for i as integer = 1 to c
    dim item as folderitem = folder.trueitem(i)
    if item<>nil and item.name = „test.txt“ then
      files.append item
    end if
  next
  for each f as folderItem in files
    f.delete
  next

6. Keep up to date

- Add update service to your app
- Use latest Xojo & Real Studio
- Use latest plugins
- Update database servers
- Verify bugs in latest version before reporting it!

7. Avoid clear text

- Drop app on text editor to see text inside.
- Encode sensitive text in your application.
- Don't embed plain text SQL commands in application code. Hackers can easily read them!
- Arbed App helps.

8. Use database permissions

- Different database access: admin for setup, worker for editing and viewer for read only access.
- Database server should only allow access from certain client IPs, Localhost, LAN, Internet.
- Let database server verify values (ranges)

9. Database Handling

- One DB Connection per Session/User.
  Keeps transaction separated.
- Use SSL for DB connection
- Never show DB error messages.
  Hackers learn DB structure from error messages.

10. Encryption / Checksums

- Use https for downloads
  HTTP/FTP SSL suported with MBS CURL Plugin
- Use SSLSocket and children.
  Verify certificate and host identity!
- Checksum data. Add a salted MD5 to verify later.
- Encrypt data. e.g. REALSQLDatabase for preferences

11. Add delays

- Add delays
- One second wait after wrong password.
- Disable User Login for 1 minute after 3 wrong logins.
- Block IPs who start lots of sessions.
- Helps fighting automated hacks.

12. Watch your memory usage

- Xojo & Real Studio applications crash on out of memory conditions.
- this crashes on low memory:
  - dim a as string = b + c
  - graphics.drawpicture p,0,0

13. Password Handling

- Don‘t store passwords if possible.
  Store hashes if needed.
- use Keychain on Mac OS X (MBS Mac Plugin)
- = vs. strcmp for passwords „a“ = „Ä“
- Help user create secure passwords.

14. Test

- Ask someone to try to hack your app
- Let newbies try app.
  Check what they input.