OWASP
Application Security Awareness

Martin Knobloch
OWASP Netherland Chapter Leader
OWASP Global Education Committee
OWASP Global Connection Committee
Sogeti Netherlands B.V.
martin.knobloch@owasp.org
+31-6 52 32 76 79

OWASP
September 16th 2010

The OWASP Foundation
http://www.owasp.org

OWASP Mobile Project:

- Collection of links to best practices
- OWASP GoatDroid
- OWASP IGoat

OWASP GoatDroid:

- Developed by Jack Mannino
- Eclipse – Android project
- Four Goat application
  - FourGoats is a location-based social network built for goats on the go.
  - Using FourGoats, you can check in at various places, earn loyalty rewards, and see what your friends are doing it, as they are doing it.
  - FourGoats will become the identity layer for the GoatDroid platform

OWASP GoatDroid:

You may encounter some of the following issues within this application:

- Client-Side Injection
- Server-Side Authorization Issues
- Side Channel Information Leakage
- Insecure Data Storage
- Privacy Concerns

- **Version 0.1.2 Beta (22 August 2011)**

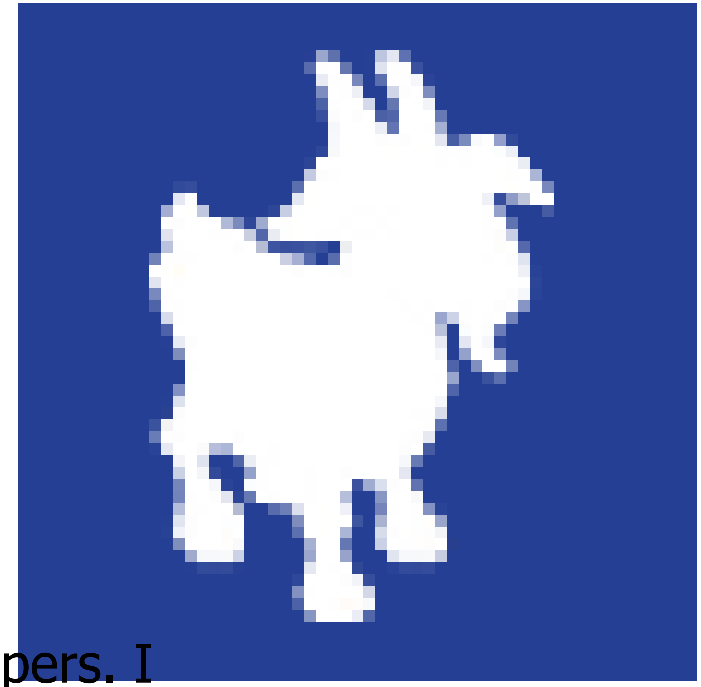
OWASP IGoat:

Developed by

Kenneth R. van Wyk

Welcome to the OWASP iGoat

learning tool



a security learning environment for iOS developers. I Goat was inspired by and loosely modeled after the OWASP WebGoat project.

As such, iGoat is a safe environment where iOS developers can learn about the major security pitfalls they face as well as how to avoid them. It is made up of a series of exercises that each teach a single (but vital) security lesson.

Version 1.2 (29 March 2012)

OWASP IGoat:

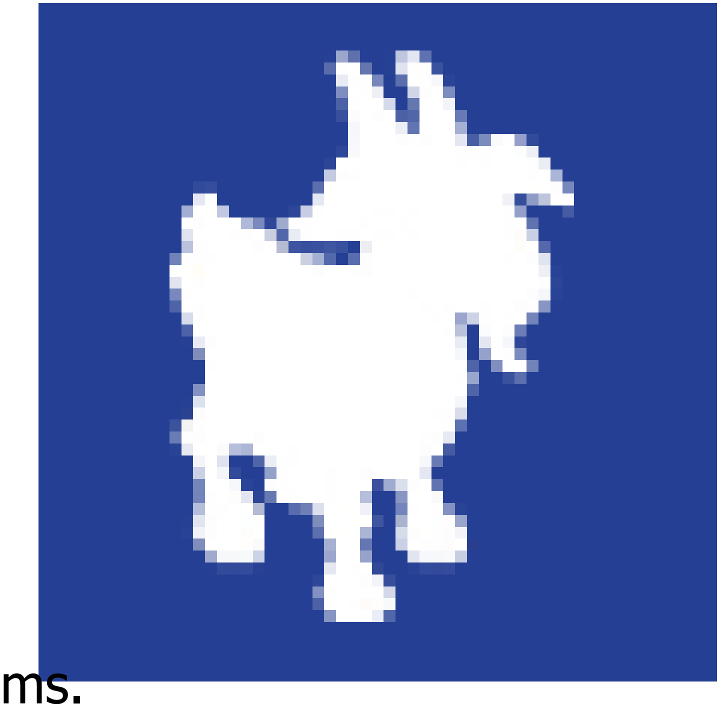
The exercises are laid out in the following steps:

- Brief introduction to the problem.
- Verify the problem by exploiting it or observing how an exploit works.
- Brief description of available remediations to the problem.
- Fix the problem by correcting and rebuilding the iGoat program.

OWASP IGoat:

Each iGoat exercise contains (at a minimum)

the following informational menu buttons:



- Exercise Plan:

Selecting this button provides you with background information on each exercise. The information describes the basic vulnerability you'll be exploiting in general terms. Next, it tells you the specifics and objectives of what you will need to do to exploit the vulnerability in the context of the exercise.

- Hints:

The hints button can be used to get some clues on how to proceed with exploiting the vulnerability in each exercise. Each time it is selected, it will give you additional information, but won't quite solve the problem for you.

OWASP IGoat:

- Remediations:

After you successfully exploit the vulnerability in each exercise, a Remediations button will appear.

When selected, it will give you some basic information on how to correct the problem in the iGoat source code.

- Solution:

When selected, the Solution button gives you complete information on how to exploit and remediate each vulnerability in each exercise

SRP

…is a secure password-based authentication and key-exchange protocol

A protocol that remains secure when:

- Attackers have complete knowledge of the protocol.
- Attackers have access to a large dictionary of commonly used passwords.
- Attackers can eavesdrop on all communications between client and server.
- Attackers can intercept, modify, and forge arbitrary messages between client and server.
- A mutually trusted third party is not available.

SRP

The SRP Project was started in 1997 at Stanford University as an authentication system for a **Java-based** webtop project. Since then, it has evolved into a full-fledged Internet-wide **Open Source** project, with developers from around the world contributing to the Project.

In addition, SRP has been deployed as a secure, free password authentication solution in commercial, non-commercial, and standalone configurations in universities, companies, and organizations worldwide

The primary goal of the SRP Project is to provide standards, technologies, and implementations that improve password security of existing protocols and applications while preserving the ease-of-use associated with passwords and integrating cleanly with these systems. SRP accomplishes these objectives because it was designed with a number of **considerations in mind**.

- **Security**
- **Convenience**
- **Openness**
- **Simplicity**

SRP

The strongest forms of authentication involve the combination of more than one authentication factor:

- What you know: Passwords, passphrases
- What you have: Hardware tokens, private keys
- What you are: Biometrics
- When combined properly, these techniques

That's it…

martin.knobloch@owasp.org

Thank you!