Secure Software Design

Andey Robins

Spring 23 - Week 10

Changes in the Second Half

Changes in the Second Half

1. Weeks 14 and 15, no class. Work on the final instead.
2. Code analysis moving to supplemental lecture
3. Dropping the Session Design assignment.
    3.1 See syllabus for point changes
4. A final “grade” will be manually entered at the end of the semester

Secure Programming

Outline

- Difficulties
- Attacks
- Common Vulnerabilities

Why is it Difficult?

Vulnerabilities are Bugs

Malicious Influence

Vulnerability Chains

Vigilance

GotoFail Revisited

All code in this section under:

```
/*
 * Copyright (c) 1999-2001,2005-2012 Apple Inc. All Rights Reserved.
 *
 * @APPLE_LICENSE_HEADER_START@
 *
 * This file contains Original Code and/or Modifications of Original Code
 * as defined in and that are subject to the Apple Public Source License
 * Version 2.0 (the 'License'). You may not use this file except in
 * compliance with the License. Please obtain a copy of the License at
 * https://www.opensource.apple.com/aps1/ and read it before using this file. */
```

```
/* The Original Code and all software distributed
   under the License are
* distributed on an 'As IS' basis, WITHOUT WARRANTY
   OF ANY KIND, EITHER
* EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS
   ALL SUCH WARRANTIES,
* INCLUDING WITHOUT LIMITATION, ANY WARANTIES OF
   MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE, QUITE ENJOYMENT
   OR NON-INFRINGEMENT.
* Please see the License for the specific language
   governing rights and
* limitations under the License.
*
* @APPLE_LICENSE_HEADER_END@
*/
```

Each call to `SSLHashSha1.update` must match an expected value to properly authenticate.

```
if ((err = SSLHashSha1.update(&hashCtx, &clientRandom)) !=
    goto fail;
if ((err = SSLHashSha1.update(&hashCtx, &serverRandom)) !=
    goto fail;
    goto fail;
if ((err = SSLHashSha1.update(&hashCtx, &signedParams)) !=
    goto fail;

// -- SNIP -- //

fail:
    SSLFreeBuffer(&signedHashes);
    SSLFreeBuffer(&hashCtx);
    return err;
```

The Problem: Structure by Syntax

```
if ((err = SSLHashSha1.update(&hashCtx, &serverRandom)) !=
    goto fail;
    goto fail;
```

Is syntactically equivalent to:

```
if ((err = SSLHashSha1.update(&hashCtx, &clientRandom)) !=
    goto fail;
}

goto fail;
```

Mitigation

Remove one of the `goto fail;` lines.

```
if ((err = SSLHashSha1.update(&hashCtx, &clientRandom)) !=
    goto fail;
```

GotoFail Commentary

Footguns

Vulnerabilities

Atomicity

Timing Attacks

Serialization

The Usual Suspects

Fixed-Width Integer Vulnerabilities

Floating-Point Precision Vulnerabilities

Examples: Underflow and Overflow

Safe Arithmetic

Memory Management

Buffer Overflow

Leaking Memory

Questions?

Next Time

- Untrusted Input
- Input Validation
- Injections