
GOOG 13

SECURING WEB APPLICATIONS

A pratical guide

@ABSTRACTJ

@SEBI2706


AeroGear

DISCLAIMER

AGENDA

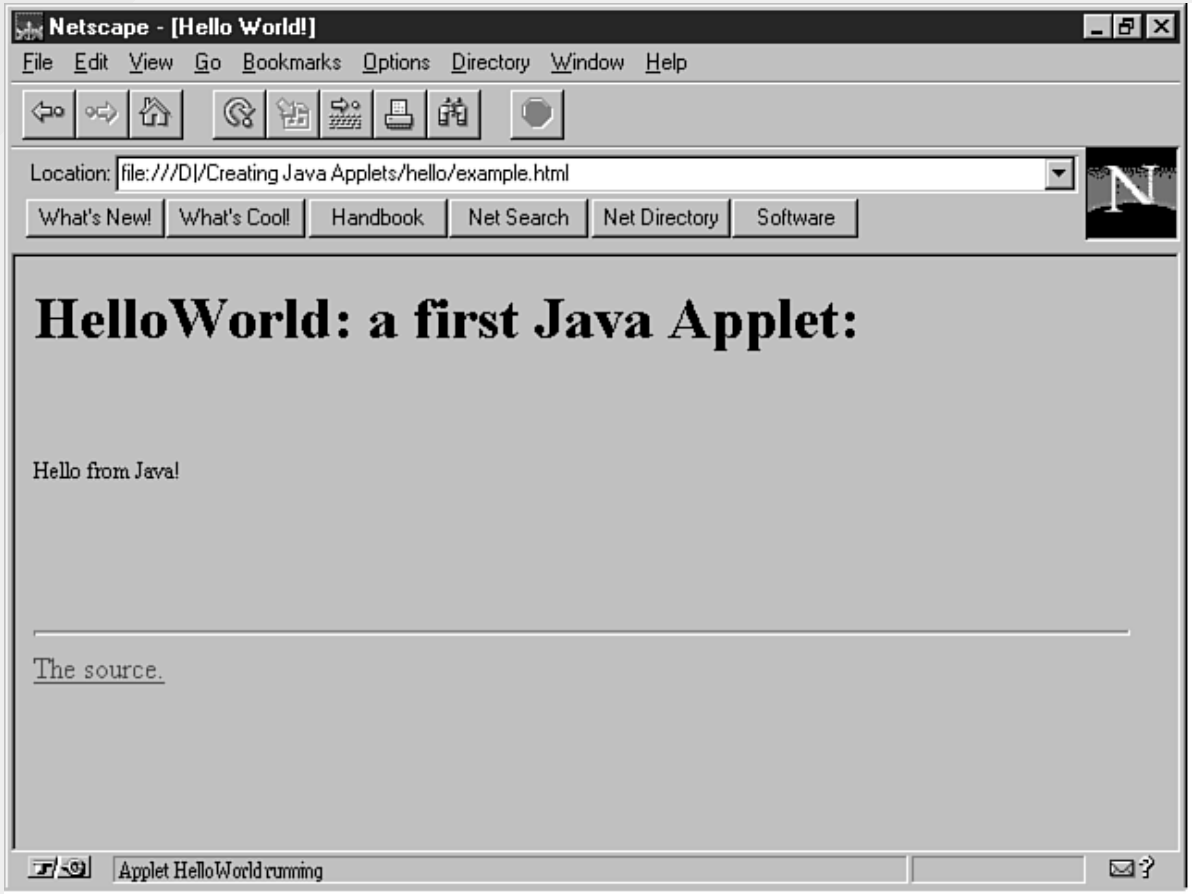
Introduction to Web security

Common vulnerabilities

Hands on


IN THE PAST

JAVA APPLETS



SERVLETS


SECURITY
FRAMEWORKS
COMPLEXITY

MAJOR THREAT


THE WEB

LOTS OF STANDARDS

W3 Standards Only (sorted by

www.w3.org/TR/tr-status-stds

W3C

Views: desktop mobile print

STANDARDS PARTICIPATE MEMBERSHIP ABOUT W3C

W3C » Standards » Standards Only (sorted by status)

STANDARDS

- Web Design and Applications
- Web Architecture
- Semantic Web
- XML Technology
- Web of Services
- Web of Devices
- Browsers and Authoring Tools
- All Standards and Drafts
- About W3C Standards

STANDARDS ONLY (SORTED BY STATUS)

Select All sorted by Status Show View

2014-03-20	CSS Namespaces Module Level 3	Recommendation *Nightly Draft*
2014-03-20	Accessible Rich Internet Applications (WAI-ARIA) 1.0	Recommendation
2014-03-20	WAI-ARIA 1.0 User Agent Implementation Guide	Recommendation
2014-03-13	Metadata API for Media Resources 1.0	Recommendation
2014-02-25	RDF Schema 1.1	Recommendation
2014-02-25	RDF 1.1 XML Syntax	Recommendation
2014-02-25	RDF 1.1 Concepts and Abstract Syntax	Recommendation
2014-02-25	RDF 1.1 Turtle	Recommendation
2014-02-25	RDF 1.1 N-Quads	Recommendation
2014-02-25	RDF 1.1 N-Triples	Recommendation

BUT NO PATTERN

HTML5

SINGLE PAGE APPS

RESTFUL ARCHITECTURE

SMARTWATCHES

JAVASCRIPT

<3

THE BROWSER

BUT IT IS ALSO HOSTILE TO SECURITY

```
beEvil();

console.log(getRandomValue());
function getRandomValue() {
  var random = new Uint32Array( 1 );
  crypto.getRandomValues( random );
  return random[ 0 ];
}

function beEvil() {
  window.crypto.getRandomValues = function( array ) {
    array[ 0 ] = 42;
  }
}
```

DEMO


SECURITY
"THE STATE OF BEING FREE FROM DANGER OR THREAT"

PEOPLE DON'T CARE

MORE THAN HALF (55%) OF INTERNET USERS ADMIT THEY USE THE SAME PASSWORD FOR MOST, IF NOT ALL, WEBSITES.

Ofcom in 2013

TOP 10 PASSWORDS

LEAKED FROM ADOBE

- 123456 - 1,911,938 users
- 123456789 - 446,162 users
- password - 345,834 users
- adobe123 - 211,659 users
- 12345678 - 201,580 users
- qwerty - 130,832 users
- 1234567 - 124,253 users
- 111111 - 113,884 users
- photoshop - 83,411 users
- 123123 - 82,694 users

SECURITY VS USABILITY

PEOPLE AVOID

eHow

Google Custom Search SEARCH

Mom | Style | Food | Tech | Home | Money | Health | Crafts | More | eHow Now

Computers | Home Theater | Mobile | Personal Electronics | Web

Featured: *Tax Time* | *Pawsitive Change* | *Smart Living* | *Investing Streamlined*

eHow » Internet » Internet Safety » General Internet Safety » How to Disable Google Malware Warnings

How to Disable Google Malware Warnings

By Diana Braun, eHow Contributor

Like Share 0 Tweet 2 Share Pin it

Computer crime has increased with malicious attacks conducted through Internet sites. These websites appear to be safe, but developers create them to illegally steal information from your computer and spread viruses. Google flags pages believed to be malware with warnings. Sites with these flags should not be opened. However, if you no longer wish to see which sites may be of harm, you can disable the Google malware warnings in the Google Chrome browser. *Have a question? Get an answer from online tech support now!*

Other People Are Reading

How to Remove a Google Malware Warning

How to Disable Google Malware Protection

PLAINTEXT PASSWORDS

See: http://plaintextoffenders.com/

THREATS TODAY

HEARTBLEED

Shellshock Continues: More Vulnerabilities Discovered

KIM ZETTER SECURITY 10.14.14 9:01 PM

THERE IS A NEW SECURITY VULNERABILITY NAMED POODLE, AND IT IS NOT CUTE

MAY 27, 2015 | BY JOSEPH BONNEAU

Logjam, Part 1: Why the Internet is Broken Again (an Explainer)

The discovery last week of another major flaw in TLS was announced, nicknamed "Logjam" by the group of prominent cryptographers who discovered it. It's getting so hard to keep track of these flaws that researchers at INRIA in France created a "zoo" classifying the attacks (which is not yet updated to include Logjam or the FREAK attack discovered in March). Despite the fact that these attacks seem to be announced every few months now, Logjam is a surprising and important finding with broad implications for the Internet. In this post I'll offer a technical primer of the Logjam vulnerability.

Logjam is actually two related but separate vulnerabilities in the way that certain common types of secure connections are established. The first is a novel active attack whereby a man-in-the-middle can force a connection to downgrade to a decades-old key exchange algorithm with well known vulnerabilities. This is a clever combination of cryptanalysis and a break in the protocol logic of TLS. The second attack, while generally known for years, was

Upgrade now: Older OpenSSL versions vulnerable to FREAK attack

Credit: iStockphoto

The OpenSSL project shared the high-severity vulnerability privately in advance as part of a post-Heartbleed strategy for security

MORE LIKE THIS

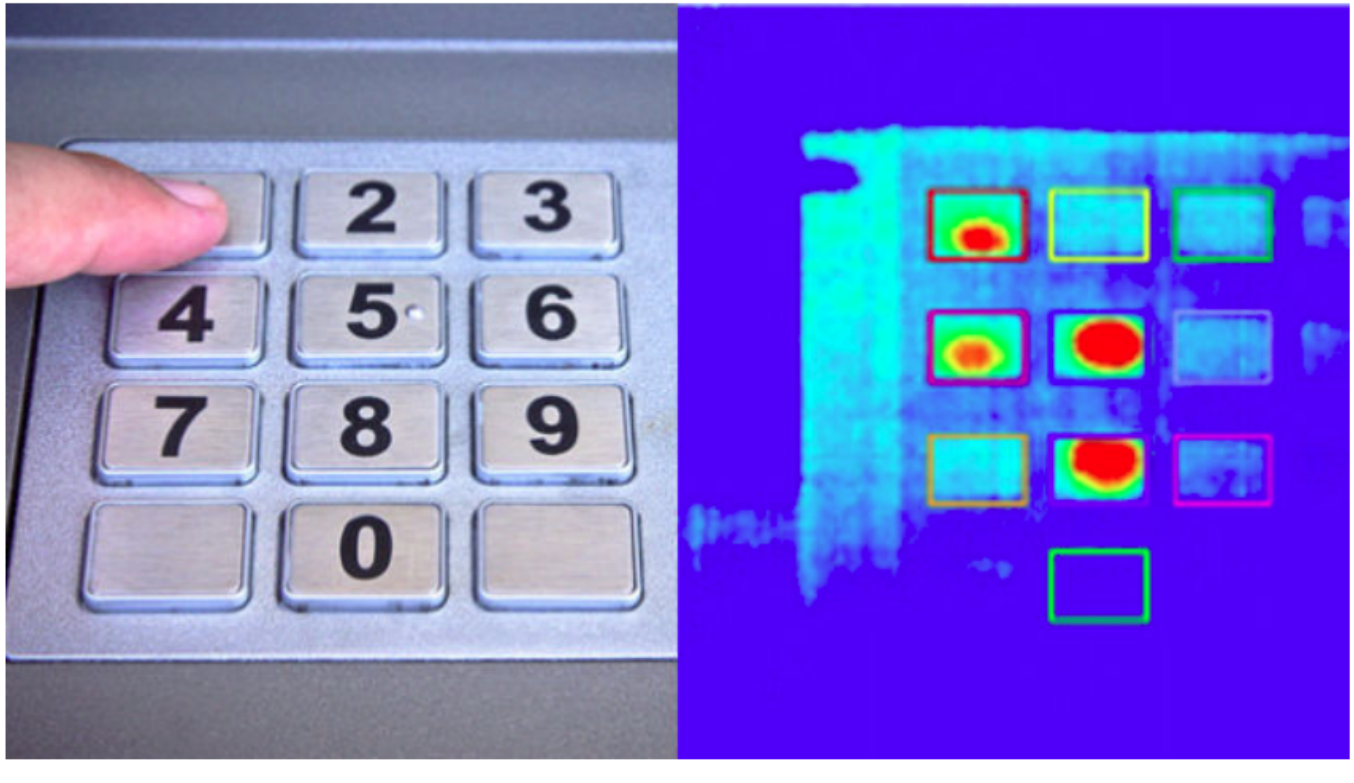
Hundreds of Android and iOS apps are still vulnerable to FREAK attacks

The state of open source security

OpenSSL patches eight new vulnerabilities

Stealing ATM PIN Numbers Using a Thermal Camera Is Too Freaking Easy

Casey Chan
Filed to: CRIME 8/17/11 6:20pm



<THE NEXT VULNERABILITY GOES HERE>

SOME PRINCIPLES

Andrew Lee Rubinger
@ALRubinger

Following

@antoine_sd @sebi2706 Don't fall asleep, baby.

Boston, MA

OWASP Top 10 – 2010 (Previous)	OWASP Top 10 – 2013 (New)
A1 – Injection	A1 – Injection
A3 – Broken Authentication and Session Management	A2 – Broken Authentication and Session Management
A2 – Cross-Site Scripting (XSS)	A3 – Cross-Site Scripting (XSS)
A4 – Insecure Direct Object References	A4 – Insecure Direct Object References
A6 – Security Misconfiguration	A5 – Security Misconfiguration
A7 – Insecure Cryptographic Storage – Merged with A9 →	A6 – Sensitive Data Exposure
A8 – Failure to Restrict URL Access – Broadened into →	A7 – Missing Function Level Access Control
A5 – Cross-Site Request Forgery (CSRF)	A8 – Cross-Site Request Forgery (CSRF)
<buried in A6: Security Misconfiguration>	A9 – Using Known Vulnerable Components
A10 – Unvalidated Redirects and Forwards	A10 – Unvalidated Redirects and Forwards
A9 – Insufficient Transport Layer Protection	Merged with 2010-A7 into new 2013-A6

CWE – CWE-602: Client-Si

cwe.mitre.org/data/definitions/602.html

Common Weakness Enumeration

A Community-Developed Dictionary of Software Weakness Types

CWSS

CWRAI

Home > CWE List > CWE- Individual Dictionary Definition (2.8)

Search by ID:

CWE List
- Full Dictionary View
- Development View
- Research View
- Fault Pattern View
- Reports
- Mapping & Navigation

About
- Sources
- Process
- Documents
- FAQs

Community
- Use & Citations
- SwA On-Ramp
- Discussion List
- Discussion Archives
- Contact Us

Scoring
- Prioritization
- CWSS
- CWRAF

Presentation Filter: --None--

CWE-602: Client-Side Enforcement of Server-Side Security

Client-Side Enforcement of Server-Side Security

Weakness ID: 602 *(Weakness Base)* **Status**

Description

Description Summary

The software is composed of a server that relies on the client to implement a mechanism that is intended to protect the serve

Extended Description

When the server relies on protection mechanisms placed on the client side, an attacker can modify the client-side behavior to bypass the protection mechanisms resulting in potentially unexpected interactions between the client and server. The consequ will vary, depending on what the mechanisms are trying to protect.

Time of Introduction

- Architecture and Design

CRYPTOGRAPHIC ALGORITHM VALIDATION PROGRAM (CAVP)

The ***Cryptographic Algorithm Validation Program (CAVP)*** encompasses validation testing for FIPS approved and NIST recommended cryptographic algorithms and components of algorithms. Cryptographic algorithm validation is a prerequisite to the Cryptographic Module Validation Program (CMVP). The CAVP was established by NIST and the Communications Security Establishment (CSE) in July 1995. All of the tests under the CAVP are handled by third-party laboratories that are accredited as Cryptographic and Security Testing (CST) Laboratories by the National Voluntary Laboratory Accreditation Program (NVLAP). Vendors interested in validation testing of their algorithm implementation may select any of the accredited laboratories.

CRYPTOGRAPHIC ALGORITHM VALIDATION TESTING SPECIFICATIONS

Below are the algorithms for which the CAVP currently

WE CAN COMMIT MISTAKES IN ANY TECHNOLOGY

#1 UNDERSCORE.JS

```
<div class="view">
  <input class="toggle" type="checkbox" <%="done" ?=""
           'checked="checked" '="" :="" ''="" %=""> />
  <label><%- title %></label>
  <a class="destroy"></a>
</div>
<input class="edit" type="text" value="<%= title %>">
```

#2 UNDERSCORE.JS

```
<div class="view">
  <input class="toggle" type="checkbox" <%="done" ?=""
           'checked="checked" '="" :="" ''="" %=""> />
  <label><%- title %></label>
  <a class="destroy"></a>
</div>
<input class="edit" type="text" value="<%- title %>">
```

WHAT'S THE DIFFERENCE?

DEMO

DEADLINES


I Am Devloper
@iamdevloper
Follow
10 lines of code = 10 issues.
500 lines of code = "looks fine."
Code reviews.
Reply
Retweet
Favorite
More
RETWEETS
4,734
FAVORITES
1,804
7:58 AM - 5 Nov 2013


MITM
Man in the middle

https://letsencrypt.org/

HSTS

HTTP Strict Transport Security

Instructs the web browser
to **interact only with HTTPS**

WILDFLY

```
<subsystem xmlns="urn:jboss:domain:undertow:1.1">
    <server name="default-server">
        <host name="default-host" alias="localhost">
                <location name="/"
            handler="welcome-content">
                <filter-ref name="hsts">
        </filter-ref></location></host>
    </server>
    <filters>
        <response-header name="hsts"
            header-name="Strict-Transport-Security"
            header-value="max-age=2592000">
    </response-header></filters>
</subsystem>
```

EXAMPLE

```
curl -I https://www.openshift.com
HTTP/1.1 200 OK
...
Strict-Transport-Security: max-age=15768000,
       includeSubDomains
Connection: keep-alive
...
```

MEET KEYCLOAK

INSTALLATION OPTIONS

STANDALONE

WILDFLY/EAP SUBSYSTEM

OPENSHIFT

DOCKER IMAGE

redhat

THANK YOU!

https://keycloak.jboss.org

https://aerogear.org