Τι λένε τα κομπιούτερς και οι αριθμοί

Chara Vassiliadou, Commercial Director

- SECURITY TESTING
- MOBILE APP TESTING
- VULNERABILITY RESEARCH
- SOURCE CODE AUDITING
- SECURE SDLC
- DIGITAL FORENSICS
- SECURITY TRAINING
- SECURITY CONSULTING

What do assessments indicate?

Top-3 issue types identified in CENSUS mobile app vulnerability assessments

Top-3 Most Common Issue Types

Insufficient data at rest protection & unintended data leakage

Misconfigured transport security & weak data in transit protection

Web API authentication/authorisation issues

Insecure Data Storage

Insufficient protection of sensitive data on the client-side

Personally Identifiable Information (e.g. credit card numbers) stored plaintext in filesystem

App assets (e.g. login credentials, session IDs) cached in a insecure manner (e.g. config files)

Device / App identifiers leaked in public directories (e.g. system caches & log files)

Improper usage of system's data security mechanisms (Keychain, Keystore, etc.)

Example attacks that can lead to extraction of device data:

Device infected with generic or targeted malware

Compromised cloud services or tethered workstations with local backups

Physically stolen device

Insecure configuration of corporate MDM (BYOD case with mixed personal/company data)

Shared devices (e.g. corporate or family tablets)

Insufficient Transport Layer Protection

Vulnerabilities that affect data in transit

Plaintext communications (e.g. HTTP and SIP)

Improper SSL certificate chain validation (self-signed certs, hostname checks, etc.)

Weak SSL/TLS security configuration (insecure protocols, weak ciphers, etc.)

Sensitive data transmitted over insecure alternative channels (e.g. SMS or Push Notifications)

Mobile device communications can be compromised from

Adversaries positioned in an adjacent network (local WiFI MitM)

Attacks at the network device or ISP/carrier level (router, proxy, cell tower, SS7, etc.)

Malware infected devices (e.g. tamper with network stack configuration and runtime)

Attacks at the web services (e.g. compromised load balancers)

Poor Authentication & Authorisation

Mobile application gateways fail to properly enforce access controls

- Lack of authentication (mostly found on hidden or dynamically defined endpoints)
- Improper data federation (e.g. Insecure Direct Object Reference bugs)
- User role transmitted from client without server validation
- Insufficient password / PIN policies

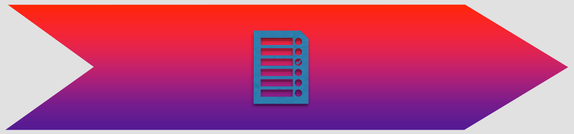
“Trusted” mobile clients can be used as an attack entry-point

- Tampered application operating in unprovisioned ways (e.g. binary patching or dynamic hooking)
- Rooted / Jailbroken devices provide enriched access over client assets (e.g. device identifiers)
- Stolen devices (e.g. valid tokens extracted from storage due to missing biometric controls)

What do assessments indicate?

Top-5 issue types identified in CENSUS web app vulnerability assessments

Top-5 Most Common Issue Types



Insufficient Throttling in Authentication Forms

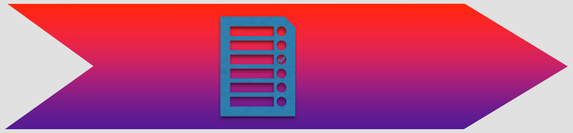
Client Side Data Validation

Broken Access Control

Server misconfiguration

Using Components with Known Vulnerabilities

Insufficient Throttling in authentication forms



Insufficient Anti-automation

Web application permits an attacker to automate a process that was originally designed to be performed only in a manual fashion, i.e. by a human web user.

Web application functionality that is often a target for automation attacks

- Application login forms
- Service registration forms
- Email forms
- Account maintenance
- Account information forms
- Forms tied to SQL database queries
- eShopping and eCommerce applications
- Web-based SMS message sending
- Online polls
- Comments forms

Client Side Data Validation

Client Side validation

Failure to validate user input on the server.

Inconsistent validation between client-side and server-side validation.

Impact of the attack

Subversion of application-specific business rules

Server-side code injection such as SQL injection, XML injection and LDAP injection

Execution of malicious code due to input, causing a buffer overflow or similar condition

Information disclosure, such as created by path manipulation or canonicalization

Denial of service caused by unconstrained or poorly formatted input.

Broken Access Control

Lack of automated detection

Lack of effective functional testing by application developers

Examples of the attack

Bypassing access control checks by modifying the URL, internal application state, or the HTML page, or simply using a custom API attack tool

Allowing the primary key to be changed to another's users record, permitting viewing or editing someone else's account

Elevation of privilege

Metadata manipulation

Force browsing to authenticated pages as an unauthenticated user or to privileged pages as a standard user.

Server misconfiguration

At any level of the application stack:
the network services / platform /web server / application server / database frameworks / custom code / pre-installed virtual machines / containers / storage

Examples of vulnerabilities

Missing appropriate security hardening across any part of the application stack

Unnecessary features are enabled or installed

Default accounts and their passwords still enabled and unchanged.

Error handling reveals stack traces or other overly informative error messages to users

For upgraded systems, latest security features are disabled or not configured securely

The security settings in the application servers, application frameworks

The software is out of date or vulnerable

Using components with known vulnerabilities

Using Components with Known Vulnerabilities

Component-heavy development patterns

Significant effort to identify the exploitability of the vulnerabilities

When you are more vulnerable

Not knowing the versions of all components (direct and nested ones)

Software is vulnerable, unsupported, or out of date (OS, web/application server, DBMS, applications, APIs, runtime environments, libraries)

Lack of periodic scanning mechanism and subscription to security bulletins

Patching is based on a time based schedule and not on risk based fashion

Software developers do not test the compatibility of updated, upgraded, or patched libraries.

Contact us

Services to keep you, one step ahead

Contact us

CENSUS
IT Security Works

EU offices:
Stadiou 33, 10559 Athens, Greece

+30 210 220 8989
+30 210 220 8990

https://census-labs.com
info@census-labs.com
@census_labs