— Odoo (the software) —

Software Security

Recent years have seen a steady increase in the digital threats faced by businesses, small and large alike. The security of business and personal data becomes more and more important every day, and the arrival of new regulation such as GDPR adds legal burden to the existing business risk.

XSS, CSRF, SQL injection, broken authentication, data leak, and so on. All kinds of security problems happen every day, even to the biggest companies. We can't stop that, but we can at least prepare for it, by carefully considering the risks, and integrating best practices into daily coding tasks.

Odoo is open source, so the whole codebase is continuously under examination by Odoo users and contributors worldwide. Community bug reports are therefore one important source of feedback regarding security.
The Odoo R&D processes have code review steps that include security aspects, for new and contributed pieces of code.

Secure by design

Odoo is designed in a way that prevents introducing most common security vulnerabilities:

- ✓ SQL injections are prevented by the use of a higher-level API that does not require manual SQL queries.
- ✓ XSS attacks are prevented by the use of a high-level templating system that automatically escapes injected data.
- ✓ The framework prevents RPC access to private methods, making it harder to introduce exploitable vulnerabilities.

See also the OWASP Top Vulnerabilities section to see how Odoo is designed from the ground up to prevent such vulnerabilities from appearing.

Independent Security Audits

Odoo is regularly audited by independent companies that are hired by our customers and prospects to perform audits and penetration tests. The Odoo Security Team receives the results and takes appropriate corrective measures whenever it is necessary.

Odoo also has a very active community of independent security researchers, who continuously monitor the source code and work with us to improve and harden the security of Odoo. Our Security Program is described on our *Responsible Disclosure* page.

OWASP Top Vulnerabilities

Here is where Odoo stands on the top security issue for web applications, as listed by the *Open Web Application Security Project* (OWASP):

- ■ Injection Flaws: Injection flaws, particularly SQL injection, are common in web applications. Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. The attacker's hostile data tricks the interpreter into executing unintended commands or changing data.

  Odoo relies on an object-relational-mapping (ORM) framework that abstracts query building and prevents SQL injections by default. Developers do not normally craft SQL queries manually, they are generated by the ORM, and parameters are always properly escaped.

- ■ Cross Site Scripting (XSS): XSS flaws occur whenever an application takes user supplied data and sends it to a web browser without first validating or encoding that content. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, possibly introduce worms, etc.

  The Odoo framework escapes all expressions rendered into views and pages by default, preventing XSS. Developers have to specially mark expressions as "safe" for raw inclusion into rendered pages.

- ■ Cross Site Request Forgery (CSRF): A CSRF attack forces a logged-on victim’s browser to send a forged HTTP request, including the victim’s session cookie and any other automatically included authentication information, to a vulnerable web application. This allows the attacker to force the victim's browser to generate requests the vulnerable application thinks are legitimate requests from the victim.

The Odoo website engine includes a built-in CSRF protection mechanism. It prevents any HTTP controller to receive a POST request without the corresponding security token. This is the recommended technique for CSRF prevention. This security token is only known and present when the user genuinely accessed the relevant website form, and an attacker cannot forge a request without it.

- Malicious File Execution: Code vulnerable to remote file inclusion (RFI) allows attackers to include hostile code and data, resulting in devastating attacks, such as total server compromise.

  Odoo does not expose functions to perform remote file inclusion. However it allows privileged users to customize features by adding custom expressions that will be evaluated by the system. These expressions are always evaluated by a sandboxed and sanitized environment that only allows access to permitted functions.

- Insecure Direct Object Reference: A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, database record, or key, as a URL or form parameter. Attackers can manipulate those references to access other objects without authorization.

  Odoo access control is not implemented at the user interface level, so there is no risk in exposing references to internal objects in URLs. Attackers cannot circumvent the access control layer by manipulating those references, because every request still has to go through the data access validation layer.

- Insecure Cryptographic Storage: Web applications rarely use cryptographic functions properly to protect data and credentials. Attackers use weakly protected data to conduct identity theft and other crimes, such as credit card fraud.

  Odoo uses industry-standard secure hashing for user passwords (by default PKFDB2 + SHA-512, with key stretching) to protect stored passwords. It is also possible to use external authentication systems such as OAuth 2.0 or LDAP, in order to avoid storing user passwords locally at all.

- Insecure Communications: Applications frequently fail to encrypt network traffic when it is necessary to protect sensitive communications.

  Odoo Cloud runs on HTTPS by default. For on-premise installations, it is recommended to run Odoo behind a web server implementing the encryption and proxying request to Odoo, for example Apache, Lighttpd or nginx. The Odoo deployment guide includes a ***Security checklist*** for safer public deployments.

- Failure to Restrict URL Access: Frequently an application only protects sensitive functionality by preventing the display of links or URLs to unauthorized users. Attackers can use this weakness to access and perform unauthorized operations by accessing those URLs directly.

  Odoo access control is not implemented at the user interface level, and the security does not rely on hiding special URLs. Attackers cannot circumvent the access control layer by reusing or manipulating any URL, because every request still has to go through the data access validation layer. In rare cases where a URL provides unauthenticated access to sensitive data, such as special URLs customers use to confirm an order, these URLs are digitally signed with unique tokens and only sent via email to the intended recipient.