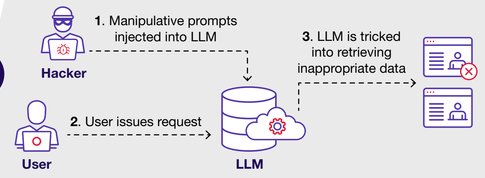
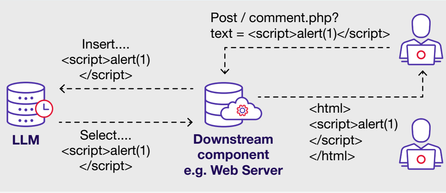
OWASP TOP 10
for LLM Applications

Prompt injection

Inject prompts to steer LLM towards desired end

1





2

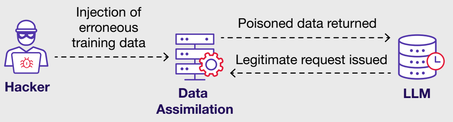
Insecure output handling

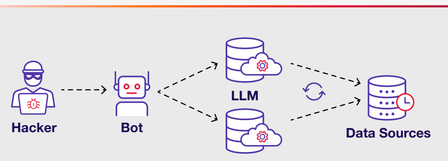
LLM output is not sanitized, enabling traditional web attacks e.g. XSS, CSRF and RCE

Training data poisoning

Training data is tampered with in order to introduce vulnerabilities into the LLM

3





4

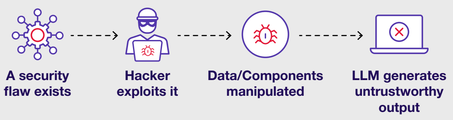
Model denial of service

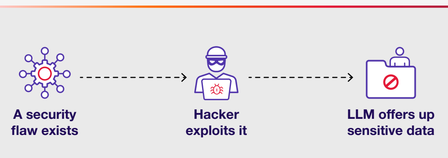
Intentionally consuming resources in order to exhaust service capacity

Supply chain vulnerability

Any vulnerabilities in the subcomponents or training data will impact the integrity of the LLM

5





6

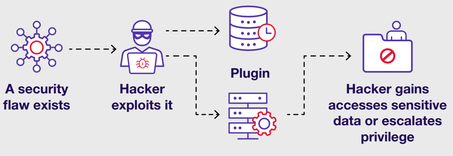
Sensitive information disclosure

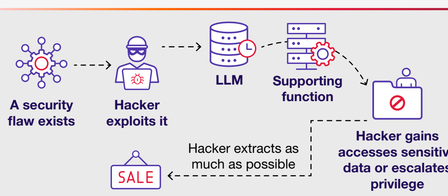
LLM may allow access to sensitive data due to insufficient built in safeguards

Insecure plugin design

Developers must use robust security measures to prevent malicious requests leading to harmful consequences

7





8

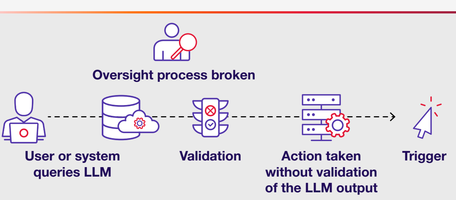
Excessive agency

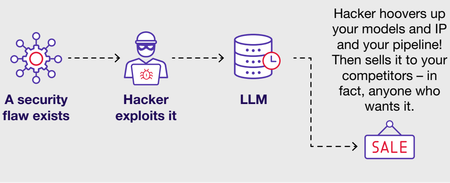
Developers must use robust security measures to limit functionality, permissions and autonomy

Overreliance

Oversight and validation must occur to ensure overreliance does not occur

9





10

Model theft

Robust security controls must be used to prevent IP theft and subsequent damage

Credit OWASP Top 10 for LLM Applications v1.1