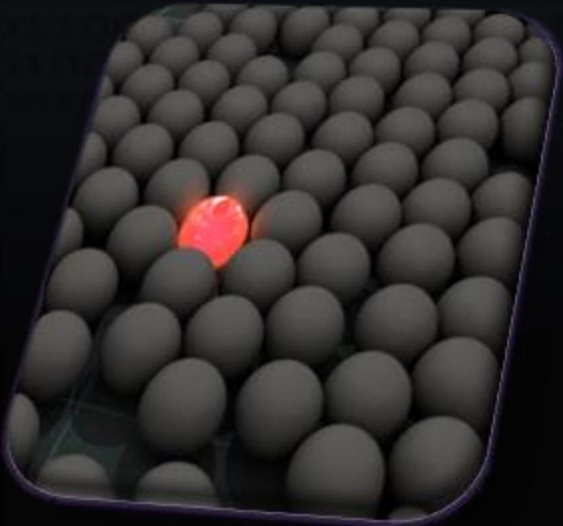
Using Components with Known Vulnerabilities

CYBER SECURITY & PRIVACY FOUNDATION

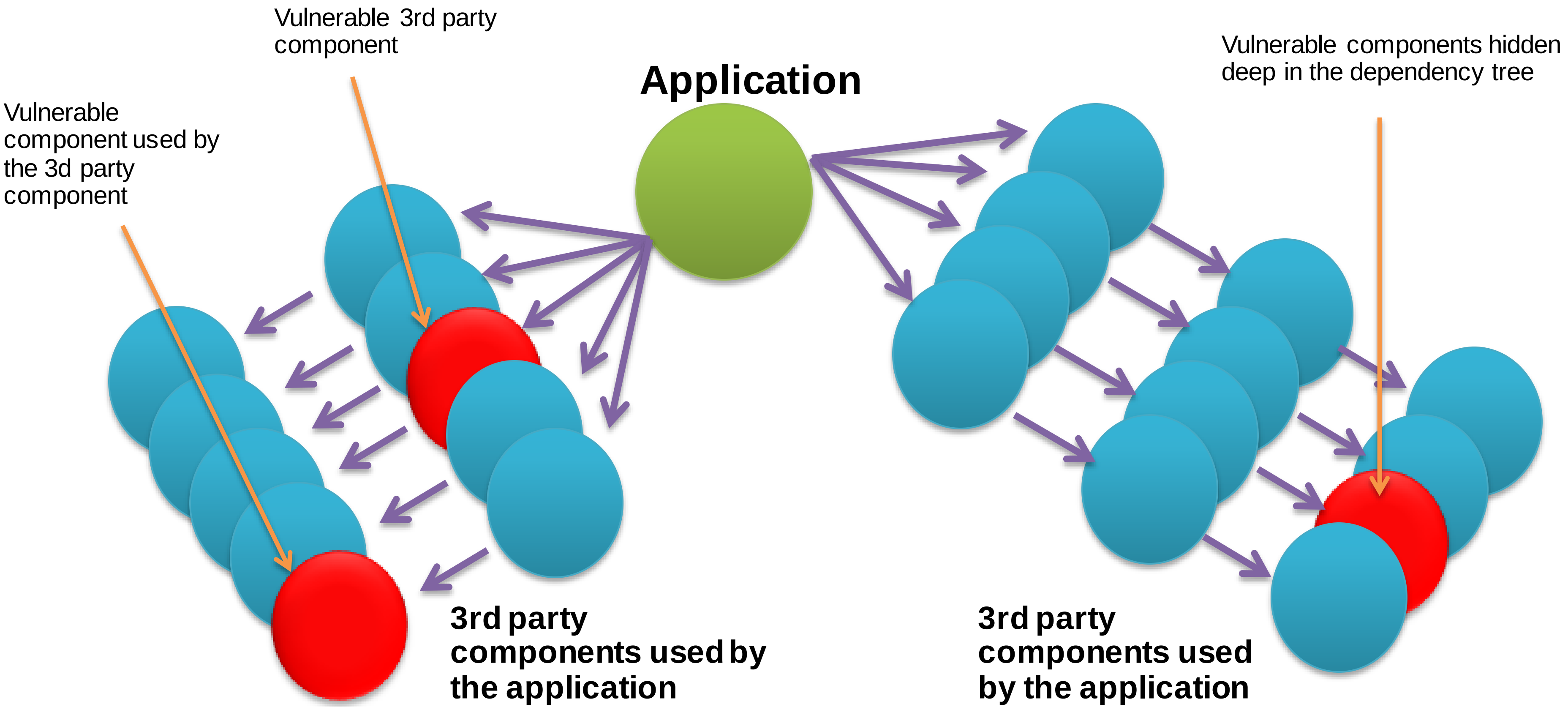
Introduction



- When creating an application, developers often make use of existing third-party libraries or frameworks to speed up the development and improve the efficiency and quality.

- But, it creates a way for vulnerable code to sneak into our secure application.

- A vulnerable components leaves the application vulnerable to hackers


Vulnerable 3rd party component
Application
Vulnerable components hidden deep in the dependency tree
Vulnerable component used by the 3d party component
3rd party components used by the application
3rd party components used by the application

In Many cases, developers don’t even know the list of third-party components used in their application.

Developers often don’t aware the components used in their application are known to be vulnerable.

Vulnerable components can cause almost any type of security risks.

Components most often have high privilege in the application, causes potential security risk(Ex: Remote code Execution).

Vulnerable Components

CVE-2005-1921:
XML-RPC for PHP Remote Code Injection Vulnerability

XML-RPC for PHP: a library implementing the XML-RPC protocol, written in PHP

XML-RPC for PHP 1.1 and prior versions are affected by a remote code-injection vulnerability

An attacker may exploit this issue to execute arbitrary commands or code in the context of the web server. This may facilitate various attacks, including unauthorized remote access.

CVE-2009-4140
Open Flash Chart 'ofc_upload_image.php' Remote PHP Code Execution

Open Flash Chart : an open source PHP library that produces some very nice-looking, interactive charts and graphs.

This vulnerability can be exploited to execute arbitrary PHP code

This vulnerable component(ofc_upload_image.php) left the OpenEMR v4.1.1 CMS vulnerable to Remote code execution.

Defence

- Identify all used third-party components(including their dependencies) and their versions.
- Monitor the security of these components in public databases, project mailing lists, and security mailing lists.
- Keep the components to date.
- Remove unused components
- Establish security policies governing component use, such as requiring certain software development practices, passing security tests, and acceptable licenses.