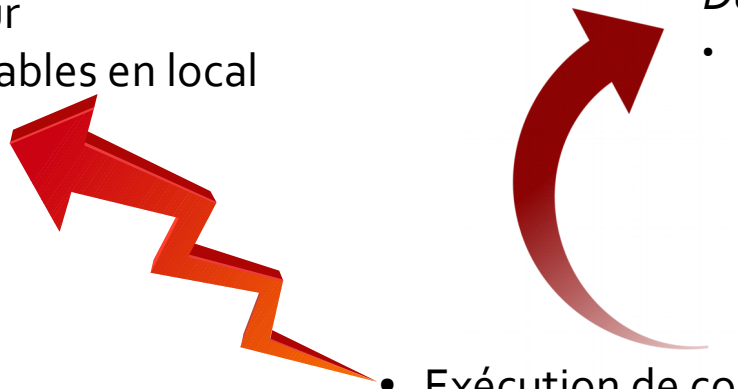
Sécurité des applications

Cross-Site Scripting (XSS) ou Injection HTML

Page web dynamique

- Serveur
  - Recette pour créer dynamiquement la page
  - Ingrédients de différentes sources
    - Base de données locale, serveur de données météo…
- Client/Navigateur
  - Scripts exécutables en local
    - JavaScript



- Exécution de code malveillant sur le navigateur

Types de XSS

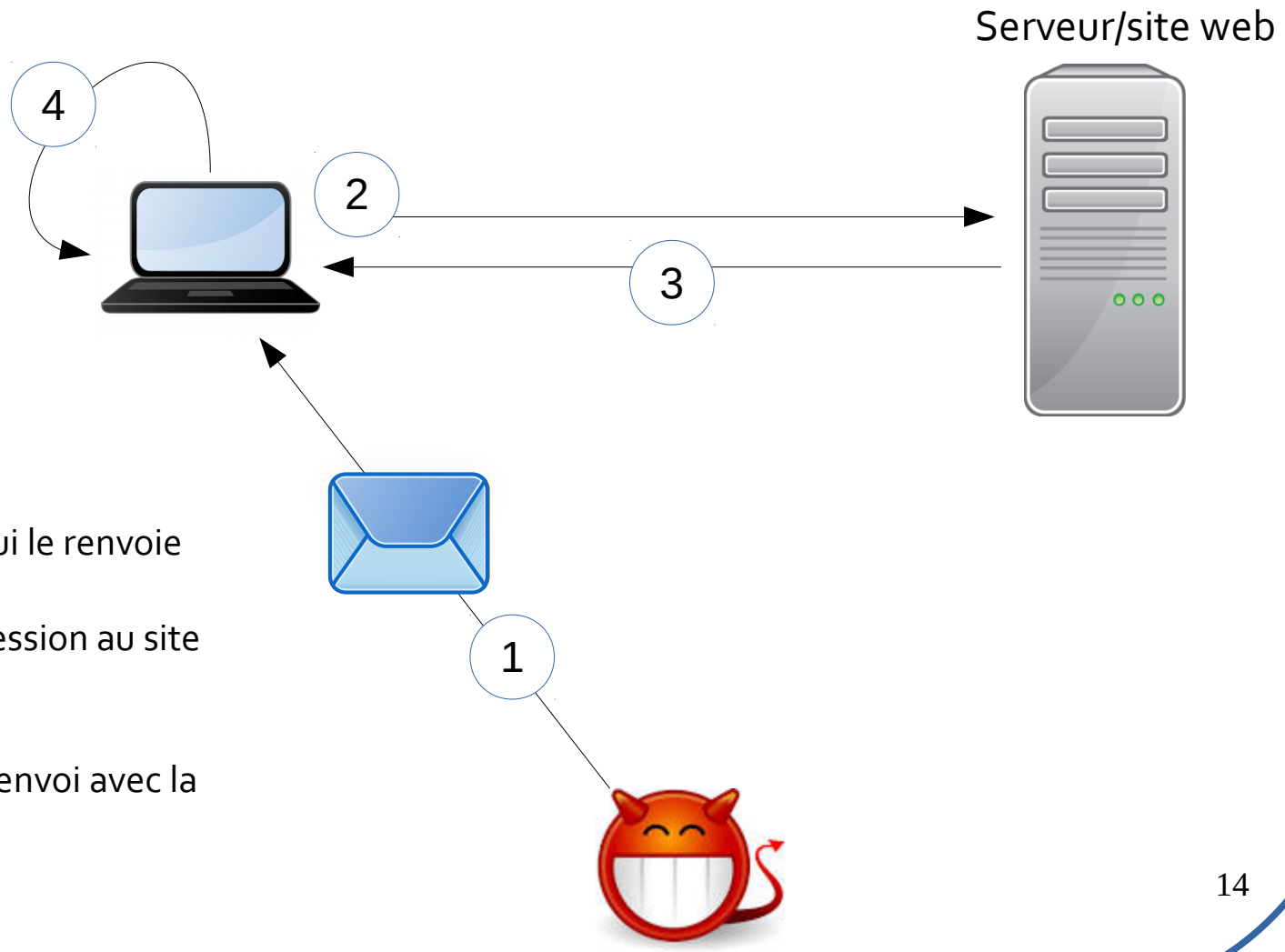
- *Reflected XSS*
- *Stored* XSS
- *Document Object Model* (DOM) XSS
  - http://www.webappsec.org/projects/articles/071105.shtml

Sécurité des applications

Cross-Site Scripting (XSS) ou Injection HTML

Types de XSS

- *Reflected XSS*



1_ Une URL non innocente

- Script soumis au serveur qui le renvoie dans la réponse
- Ex : envoyer le cookie de session au site du pirate

2_ Clic : requête go !

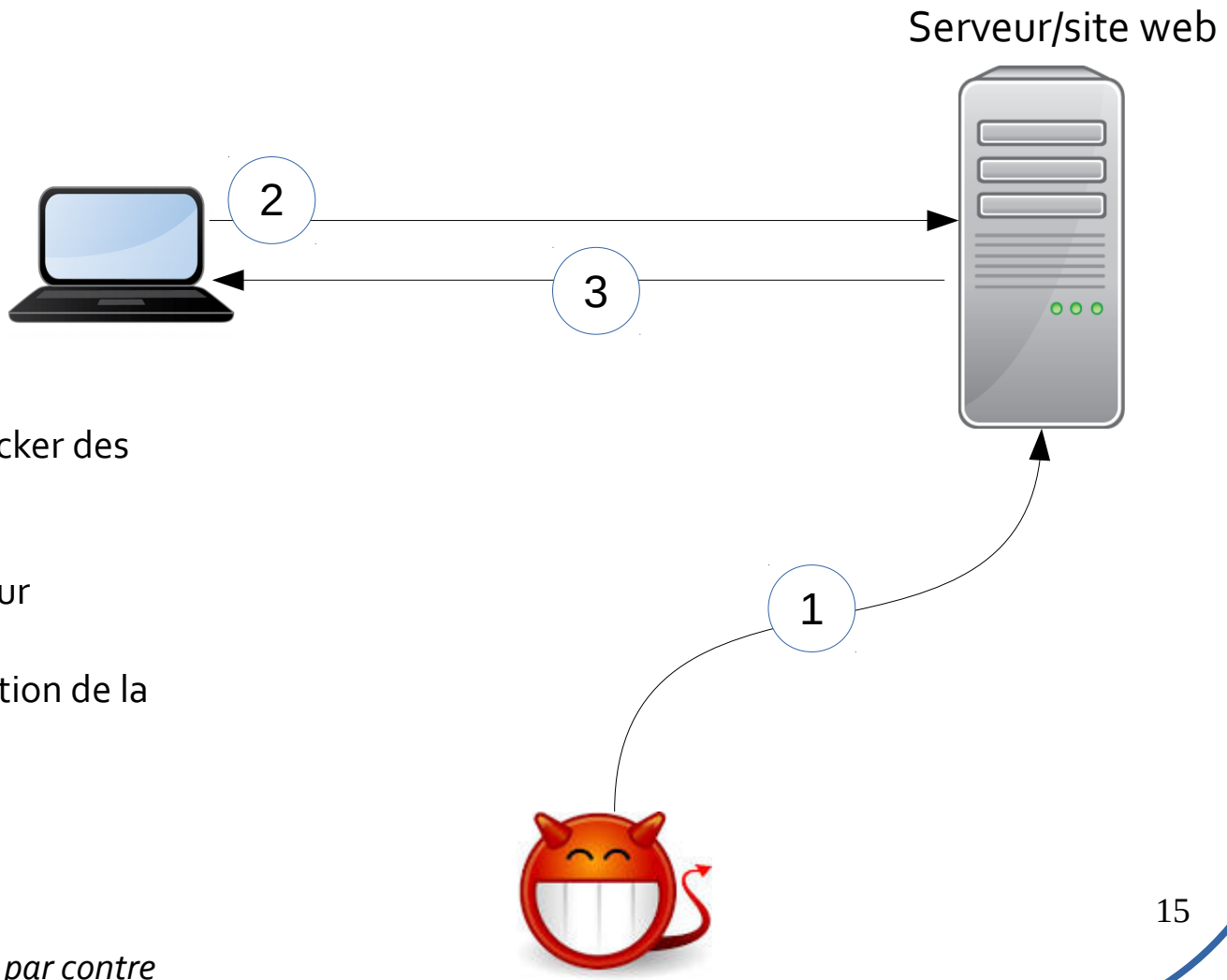
3_ Création de la réponse et renvoi avec la *payload*

4_ Exécution de la *payload*

Sécurité des applications

Cross-Site Scripting (XSS) ou Injection HTML

Types de XSS
- *Reflected XSS*
- *Stored* XSS



Exploitation de la possibilité de stocker des données sur le serveur* !
1_ Stocker la *payload* dans la DB
2_ Appel de la page par un utilisateur légitime
3_ Chargement de la page et exécution de la *payload*

** : sans les désinfecter par contre*

Sécurité des applications

Cross-Site Scripting (XSS) ou Injection HTML

```
<html>
    <head>
        <title>
            Ta page
        </title>
    </head>
    <body>
        <?php echo "Bonjour" .$_GET['nom'] ; ?>
    </body>
</html>
```

http://www.monsite.com/index.php?nom=Jason

http://www.monsite.com/index.php?nom=Jason

http://www.monsite.com/index.php?nom=M<script>
alert(document.cookie.toString())</script>

Sécurité des applications

Cross-Site Scripting (XSS) ou Injection HTML

Cible	Risques	Conséquences
• Utilisateur final, client du site d'origine • Site web d'origine	• Vol de données privées (cookies…) • Détournement de formulaires vers un autre site	• Usurpation d'identité • *The sky is the limit…* • Perte de réputation

Sécurité des applications

Défenses contre le *Cross-Site Scripting* (XSS)

- Injection HTML => Neutralisation des caractères spéciaux : technologie et langage
  Ex : PHP fournit *htmlentities*() et *htmlspecialchars*()
  - Caractères spéciaux => entité HTML : < et &lt
  - Lisibilité ?
- Cadres dans les pages web (<iframe> et <frame>)
  - Plusieurs pages sur la page : transparent pour l'utilisateur final
  - Domaines différents et politique JavaScript de restriction au domaine en cours
- Contrôle des éléments « exécutables »
  - Injection de balises JavaScript
  - Déclenchement d'actions au chargement d'une image
- ...