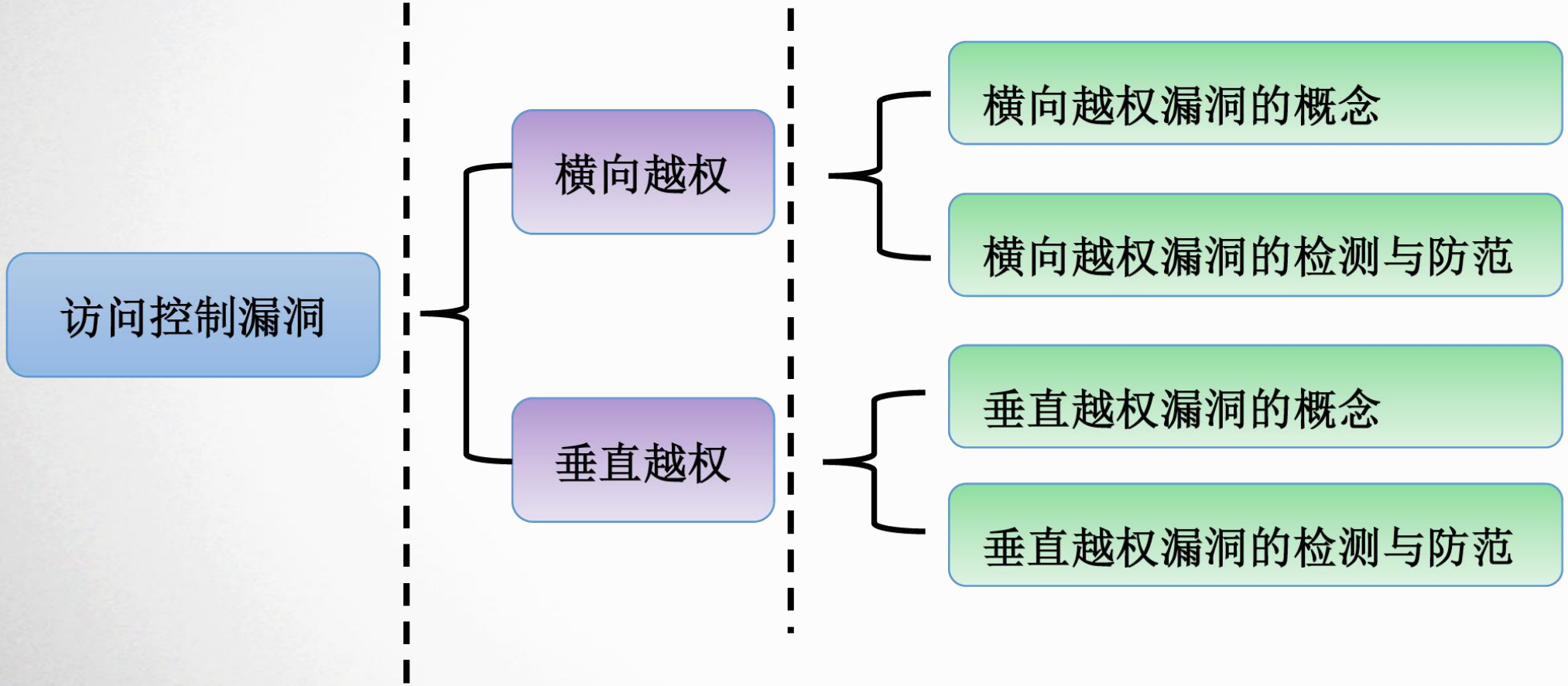
培训教育
Training Services

CISP-PTE

Web 安全基础(6) – 访问控制漏洞

主讲：

访问控制漏洞



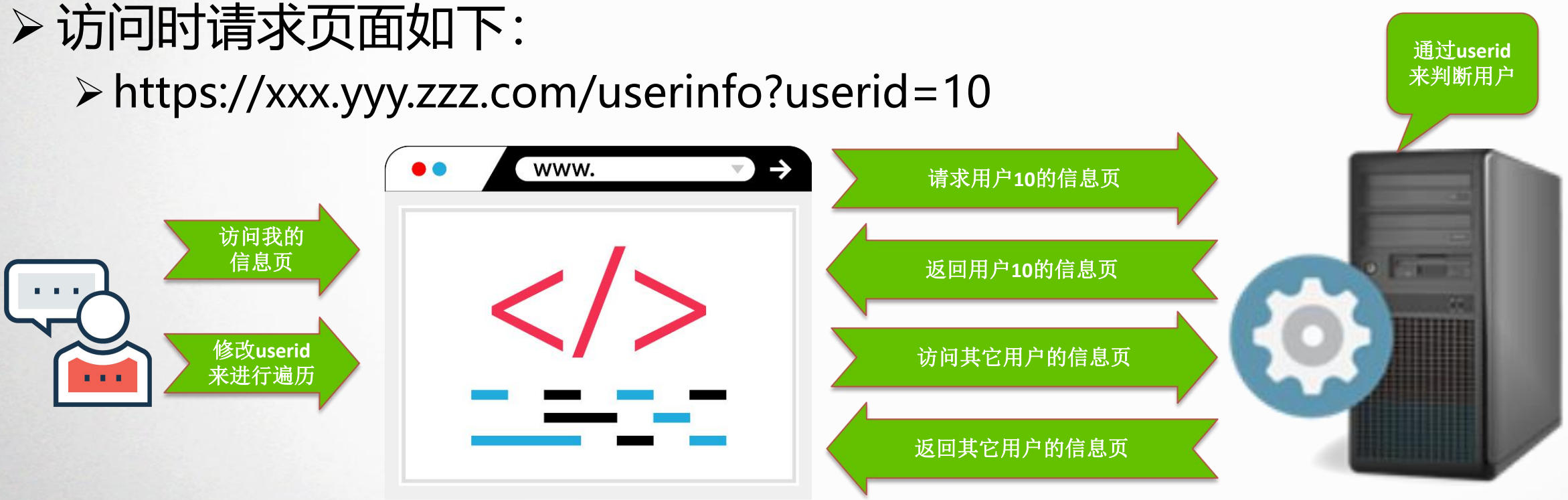
横向越权

横向越权

- 通过本知识域，我们会：
  - 横向越权漏洞的概念
    - 了解横向越权漏洞的基本概念
    - 了解横向越权漏洞的形式
  - 横向越权漏洞的利于与防范
    - 了解横向越权漏洞对网站安全的影响
    - 掌握横向越权漏洞的测试和修复方法

横向越权漏洞概念图 - 1

- 用户可以访问自己信息页
  - 可以看到姓名，身份证号，手机号等信息
- 访问时请求页面如下：
  - https://xxx.yyy.zzz.com/userinfo?userid=10



横向越权漏洞概念图 - 2

- 用户可以修改自己密码
  - 先使用邮箱等验证后，进入修改密码页面
- 最后修改用户密码的post参数如下：
  - userid=10&password_new=xxx&password_confirm=xxx

横向越权漏洞概念

- 越权漏洞属于逻辑漏洞。利用业务逻辑在程序中体现时，仅仅限制于用户点击。
- Web应用程序接收到用户请求，修改某条数据时，没有判断数据的所属人，或者在判断数据所属人时从用户提交的表单参数中获取了用户信息，导致攻击者可以自行设置用户，修改不属于自己的数据
- 只要是权限验证不是使用cookie来验证，都有可能发生横向权限漏洞
- Web应用程序接收到用户请求，修改某条数据时，没有判断数据的所属人，或者在判断数据所属人时从用户提交的表单参数中获取了userid。导致攻击者可以自行修改userid修改不属于自己的数据。所有的更新语句操作，都可能产生这个漏洞。

横向越权漏洞实例

- http://mydvwa:8080/vulnerabilities/permission/
- 正常来说，获取的是我自己信息
- 通过抓包发现获取信息时会传递用户名
  - 我们可以尝试更改用户名来获取其它用户的信息

横向越权漏洞实例

- http://mydvwa:8080/vulnerabilities/permission/
- 正常来说，获取的是我自己信息
- 通过抓包发现获取信息时会传递用户名
  - 我们可以尝试更改用户名来获取其它用户的信息

横向越权漏洞对网站安全的影响

- 以其它用户的身份进行操作
  - 查看 / 遍历内容
  - 更改信息
    - 邮箱
    - 手机号
  - 发表文章
- 所带来的影响
  - 更改确认邮箱 / 手机号，可以进行支付操作
  - 查看其它用户信息，收集隐私信息
    - 姓名、身份证号、邮箱、电话
  - 发表敏感文章
    - 使用公众号发表带有敏感信息的文章

漏洞代码

```
int userid=Integer.valueOf( request.getParameter("userid"));   -- 从用户输入的参数里确认用户ID
String email=request.getParameter("email");
String tel=request.getParameter("tel");
String realname=request.getParameter("realname");
String pass=request.getParameter("pass");
JdbcConnection conn = null;
try {
    conn = new JdbcConnection();
    Object[] params = new Object[5];
    params[0] = email;
    params[1] = tel;
    params[2] = realname;
    params[3] = pass;
    params[4] = userid;
    final String sql = "update user set email=?,tel=?,realname=?,pass=? where userid=?";
    conn.execUpdate(sql,params);
    conn.closeConn();
}
```

查找横向权限漏洞

- 查看任何传递用户信息的参数
  - 只要是用户id，用户名等是以参数方式传递，就有可能有风险
- 查看特权内容，是否进行了权限管理
  - 每个人博客等，即使是私有的：
    - 显示我可看列表进行了权限设置
    - 查看详情时，根据内容ID来进行判断

漏洞代码修复方案

```
int userid=Integer.valueOf(GetUseridFromCookie(request));   -- 从cookie中确认用户ID
String email=request.getParameter("email");
String tel=request.getParameter("tel");
String realname=request.getParameter("realname");
String pass=request.getParameter("pass");
JdbcConnection conn = null;
try {
    conn = new JdbcConnection();
    Object[] params = new Object[5];
    params[0] = email;
    params[1] = tel;
    params[2] = realname;
    params[3] = pass;
    params[4] = userid;
    final String sql = "update user set email=?,tel=?,realname=?,pass=? where userid=?";
    conn.execUpdate(sql,params);
    conn.closeConn();
}
```

如何修复横向权限漏洞

- 此类漏洞，很多时候是业务分析到程序设计时产生的。因此很多时候需要更改程序逻辑。
  - 用户id，用户名等禁止通过参数来传递，直接取Cookie里的值
  - 私有信息访问时需要验证用户身份
    - 隐藏的博客等，需要验证用户身份，而不只是通过内容ID来取信息
    - 在数据库取数据时，需要验证
      - 原来语句：select * from blogs where blog_id = xx;
      - 休整后：select * from blogs where blog_id = xx and owner = yy;
- 要是短时间很难更改整体逻辑，可以通过混淆参数方法来进行防御
  - 用户ID等使用MD5码等，很难进行遍历

垂直越权

垂直越权

- 通过本知识域，我们会：
  - 垂直越权漏洞的概念
    - 了解垂直越权漏洞的基本概念
    - 了解垂直越权漏洞的种类和形式
  - 垂直越权漏洞的检测与防范
    - 了解垂直越权漏洞对网站的影响
    - 掌握垂直越权漏洞的测试和修复方法

垂直权限漏洞的概念

- 垂直权限攻击又叫做权限提升攻击。其原理是由于Web应用没有做权限控制，或仅仅在菜单上做了权限控制，导致恶意用户只要猜测其他管理页面的URL，就可以访问或控制其他角色拥有的数据或页面，达到权限提升的目的。
- 后台管理页面一般只允许管理员访问，如果普通用户可以访问，就存在向上越权漏洞。
- 解决向上越权是比较容易处理的事情，如果管理员表与普通用户表是同一张数据库表，就必须要存在权限验证字段，权限验证字段用来区分是否为管理员。

垂直权限漏洞 – 例1

```
public void doFilter(ServletRequest req, ServletResponse res,
FilterChain filter) throws IOException, ServletException {
    HttpServletRequest request = (HttpServletRequest) req;
    HttpServletResponse response = (HttpServletResponse) res;
    User user = (User) request.getSession().getAttribute("user");  // 从Cookie验证用户
    if(user==null){
        request.getRequestDispatcher("/").forward(request, response);//  跳转操作
    }
    boolean flag = user.getIsAdmin();
    if (flag) {
        filter.doFilter(request, response);
    } else {
        request.getRequestDispatcher("/").forward(request, response);//  跳转操作
    }
}
```

垂直权限漏洞 – 例2

```
<tr>
  <td><a href="/user.jsp">管理个人信息
</a></td>
</tr>
<%if (power.indexOf("administrators")>-1){%>
  <tr>
    <td><a href="/userlist.jsp">管理所有用户
</a></td>
  </tr>
<%}%>
```

垂直权限漏洞 – 例3

- 直接访问管理员页面
  - 权限控制在前台做。
  - 检查用户是不是admin，不是就进行跳转到登陆页面
  - 使用抓包工具，不接收跳转语句，就可以一直访问管理员页面

垂直越权漏洞对网站的影响

- 垂直权限漏洞，一般是直接访问业务管理员权限
  - 可能会看到全部用户信息
  - 可能更改全部通告信息（注入广告信息）
  - 可能更改商品价格
  - 可能更改订单信息
- 只要是能拿到业务管理员权限，可能会对业务造成很大影响
  - 价格改动引起的低价卖商品而赔钱 – 金钱影响
  - 虚假通知引起的多人被骗 – 信誉影响
  - 更改订单引起的订单丢失 - 信誉 / 金钱双重影响
  - 用户信息泄漏 – 隐私泄露 / 影响信誉

垂直越权漏洞的修复方法

- 在每个页面的加载之前进行权限验证
  - 进行服务器验证 – 不能在前台验证
  - 验证时，从session获取对应的用户信息
  - session中用户信息存放在服务端，用户不能修改
  - 对每个敏感页面（管理员页面）都进行验证