技术与经验分享研讨会　第一场（下）

CTF-Web 预备知识

东北大学 NEX 信息安全创新团队
2024年8月24日

计算机基础课程

CS fundamental courses

- 数据结构和算法：堆栈、队列、树（AVL）、图（Call graph）
- 计算机组成原理：CPU 架构、侧信道攻击
- 计算机网络：TCP/IP、网段与路由、HTTP、协议分析
- 操作系统：环境、进程、文件、Windows、Linux

引入
Introduction

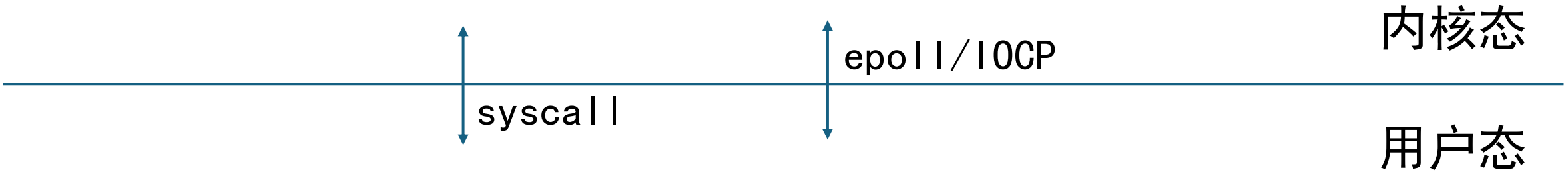
用户打开浏览器，输入网址，直到看到网页内容；

总共经历哪些过程？

引入
Introduction

1. 键鼠产生中断事件，由 CPU 切换优先级处理

3. 进行进程创建、文件访问（注册表）、套接字等操作

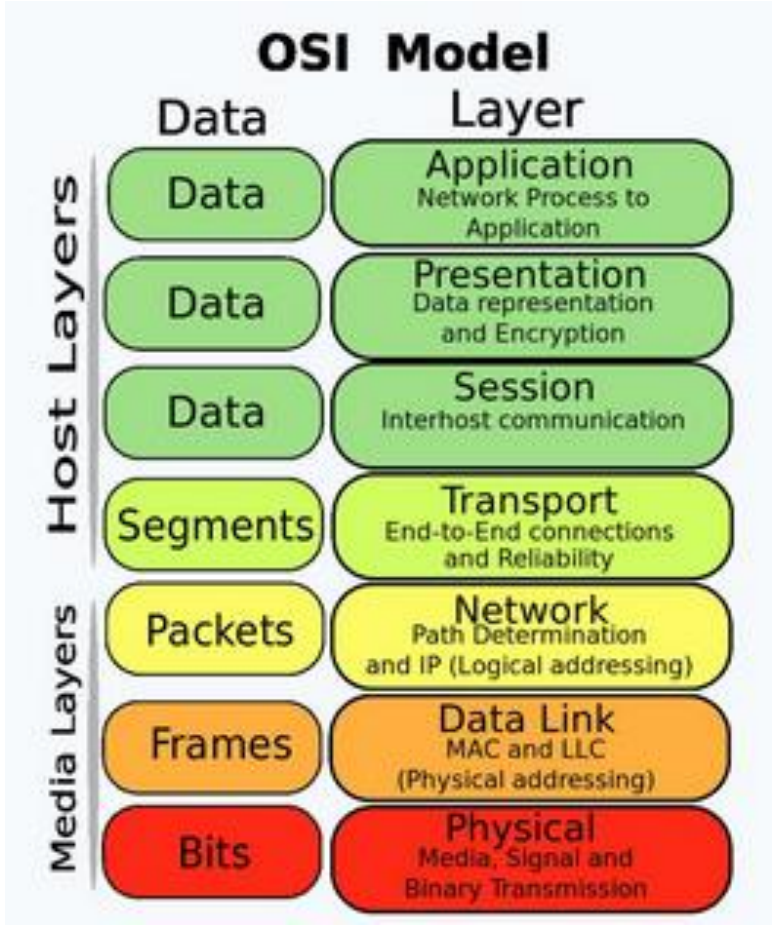


2. 各种 event 传递至用户 dwm（explorer.exe）

4. 各种 event 传递至 msedge.exe，进行对应操作

引入
Introduction

- 二层：ARP 或 ICMPv6-RA
- 三层：IP Fragmentation/NAT
- 四层：TCP 三次握手/UDP
- 七层：DHCP/DNS/HTTP



进程
Processes

- Procmon / procexp
- Autoruns64 / Services / Task Scheduler
- Pipes / Messages
- netstat -na

- ps –ef / strace / lsof / env
- /proc/
- systemd / cron
- file / vsock
- netstat –natp / ss -natp

权限
Permissions

- **NTFS File Permissions**

- **User Privileges:**

whoami /priv
incognito list_tokens -u

- **NTLM: Mimikatz**

- **UNIX Permissions**

- **SUID/SGID:**

find / -type f -perm -4000 -ls 2>/dev/null

- **cgroup**

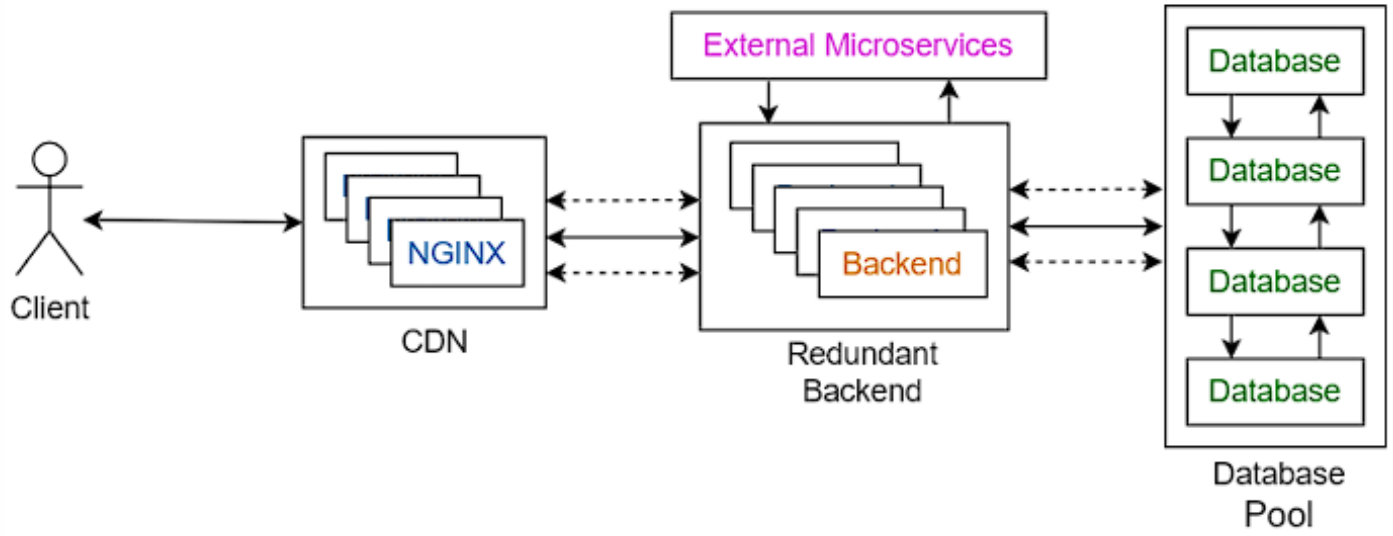
- **File Capabilities:**

getcap -r / 2>/dev/null

- **/etc/shadow**

软件部署
Software deployment

- 中间件：Nginx / Apache2
- 数据库：SQLite / MySQL / PostgreSQL / SQL Server / Oracle DB / MongoDB
- 缓存：Memcached / Redis
- 脚本语言：PHP / ASP(X) / JSP(X)
- 后端语言：.Net / Java / Python / Node.js / Go / Rust / C / C++
- 交互协议：HTTP / JSON / JWT / HMAC签名



PHP 部署

PHP deployment

- CLI：直接从终端运行 PHP 文件（没有 $_GET、HTTP_HOST 等变量）
- CLI-SERVER：启动 PHP 内置的 HTTP 服务器（只能服务固定的 PHP 文件）
- Apache module（mod_php）：以 mod 的形式在 Apache 中运行 PHP 代码
- CGI：以 CGI 的传统形式运行 PHP 代码（若 AllowOverride 则可以“提权”）
- FastCGI（php-fpm）：提供通用的 FastCGI 接口
  （若 SSRF (tcp:9000 unix:sock) 则可以“提权”）

Java 部署
Java deployment

- **Run WAR/JSP：Tomcat、JBoss、Jetty、Glassfish、WebLogic**

/manager/html 弱密码，AJP 任意文件包含，webapps/ 写入，web.xml 泄露

- **Run JAR：java -jar ； 整合 Springboot（Tomcat）、Netty**

heapdump 泄露、routing 泄露、
env 泄露、POST /env RCE

S2-XXX

泄露 HTTP Session

- **常见组件：Spring Boot Actuator、Apache Shiro、Struts2、Log4j2、Druid**

未授权访问、filter 绕过、反序列化 RCE

Log4Shell

数据库
Databases

- SQLite：VACUUM INTO、ATTACH DATABASE 写文件
- MySQL：错误配置 secure_file_priv 时，可以通过 load_file()、DUMPFILE 读写文件，甚至可以通过 UDF plugin RCE
- PostgreSQL：高权限用户通过 COPY FROM 直接 RCE
- SQL Server：高权限用户通过 xp_cmdshell 直接 RCE
- H2 Database：注入后直接执行 Java 代码 RCE
- Oracle DB：低版本或高用户可以绕过限制执行 Java 代码
- MongoDB：错误使用 query() （传入数组）时造成 NoSQL 注入

CRLF 注入
CRLF Injection

- HTTP CRLF Injection：常见于 SSRF 漏洞，可注入任意 HTTP Header，甚至新的 POST 请求
- HTTP Request Smuggling：利用前后服务器对 HTTP 请求的解析差
  Content-Length v.s. Transfer-Encoding
- Memcached：搭配 PHP 时可注入任意命令
- Redis：非法命令时不退出，可借由 HTTP 协议的 SSRF 进行注入
  可写任意文件 (/var/spool/cron)；可以主从复制 module load RCE

中间协议
Middleware protocols

- JSON：FastJSON 漏洞、\u 绕过 WAF、duplicate entries、numeric
- JWT：none algorithm 漏洞、kid 注入、RS256 与 HS256 混淆漏洞
- 随机数可预测：使用 timestamp、可计算的 seed 产生安全密钥
  php rand()、Node.js V8 rand()、Python MT19937 random.rand()
- Apache ActiveMQ 反序列化、WebSocket 走私欺骗
- SSTI（SpEL、EJS、Jinja2）、Flask Debug PIN

练习
Practice

1. 最外层使用 Apache2 反代并配置 https 证书（443 端口）；
2. 使用 Nginx + PHP-FPM 搭建一个 PHP 网站（8080 端口）；
3. 使用 PHP 的 curl 扩展反代内部网站;
4. 使用 SpringBoot 打包一个 JAR 包，开设 HTTP 服务，将请求参数直接传入后端数据库，模拟 SQL 注入（开启 Actuator、开启多行执行功能、8083 端口）；
5. 在本机搭建 PostgreSQL 数据库;
6. 通过 443 端口 https 访问成功 RCE。

感 谢 您 的 聆 听

Thank you for your listening