

漫谈防火墙技术



童进 @IDF实验室

主要内容

什么是防火墙

防火墙技术的发展

目前的主流产品形态

架构原理与转发流程

应用识别原理与IPS技术

APT防护

未来展望

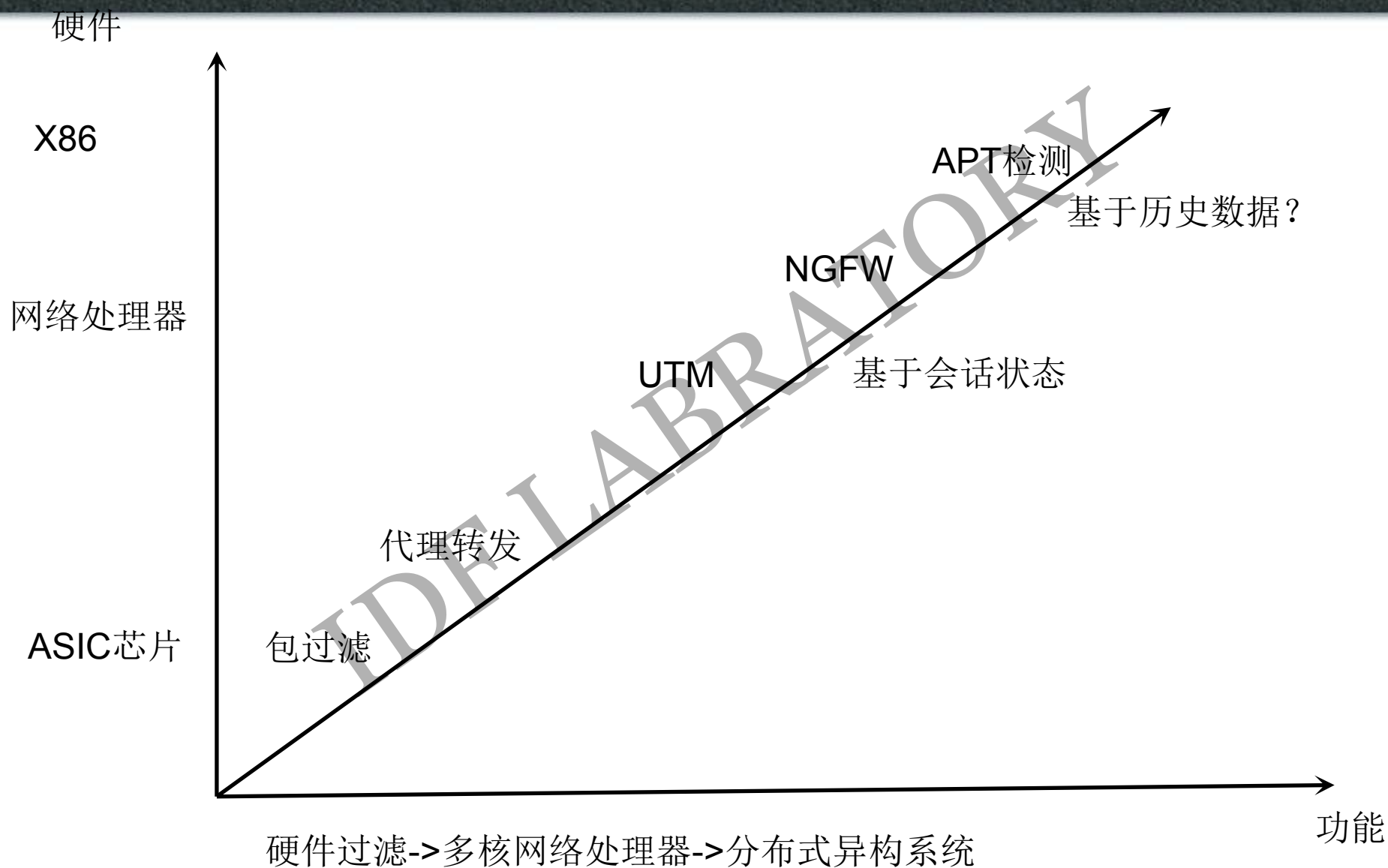
IDF LABORATORY


什么是防火墙

- 防火墙（**Firewall**）是用来加强网络之间**访问控制**的特殊**网络设备**，用于构造相对安全的子网环境。
- 本质：隔离内外网络，对进出信息流进行控制，是一种被动防御的保护装置。
- 对于同属于企业级、部署在网络边界上的三层交换机，防火墙与它的区别是：

三层交换机	防火墙
保障网络数据的“通”	阻断任何非法报文“不通”
二/三层转发技术	DPI技术
组网配置	安全策略
Switch芯片	x86/NP/ASIC(ACL/vpn加速等)
--/--	审计与日志
C-Plane & D-Plane	

防火墙技术的发展





包过滤：

方式：逐包扫描，无会话状态

架构：硬件ASIC芯片(ACL规则)

原理：

1.基于zone (TRUST/UNTRUST/DMA)

本质是基于接口，将接口加入ZONE

2.基于ip和port

代理转发：

方式：对应用进行代理转发

架构：类似代理服务器，软件处理

UTM/NGFW：（矮胖子与高瘦子）

方式：逐包扫描，基于会话状态

架构：ASIC芯片/NP处理器/x86 分布式/集中式

原理：

DPI+DFI

功能；all in one

1.安全业务：

VPN/ALG/应用识别/用户管理/URL过滤/上网行为控制/IPS/AV

2.其它业务：

负载均衡/QoS

PPPoE/AAA

NAT/DHCP/DNS代理

IGP/BGP协议的支持

SSL代理（审计需求）

目前主流厂商

Gartner Magic Quadrant for Enterprise Network Firewalls 2014



		新建	并发	IPS吞吐
CheckPoint	高性能	600K	70M	40~110Gbps
	中低端	25K	1.2M	0.3~2Gbps
Palo Alto	高性能	720k	24M	60~100Gbps
	中低端	15K	250K	0.2~0.5Gbps
Fortinet	高性能		4~132M	
	中低端	250K	12M	4.3Gbps
Hillstone	高性能	2.4M	120M	
	中低端	4~9K	200~600k	

架构原理与基本流程

硬件架构

软件架构

重要概念：

Session

Policy

转发流程

IDF LABORATORY

硬件架构

硬件芯片

x86

Cavium

ASIC

混合（分布式设备，不同的扩展卡）

分布式 vs 集中式

优势：便于扩展/高性能

难点：跨板转发/全局状态同步

软件架构(Cavium----集中式设备)

Zero copy ---- 网络IO

多核 ---- SMP , 并行处理

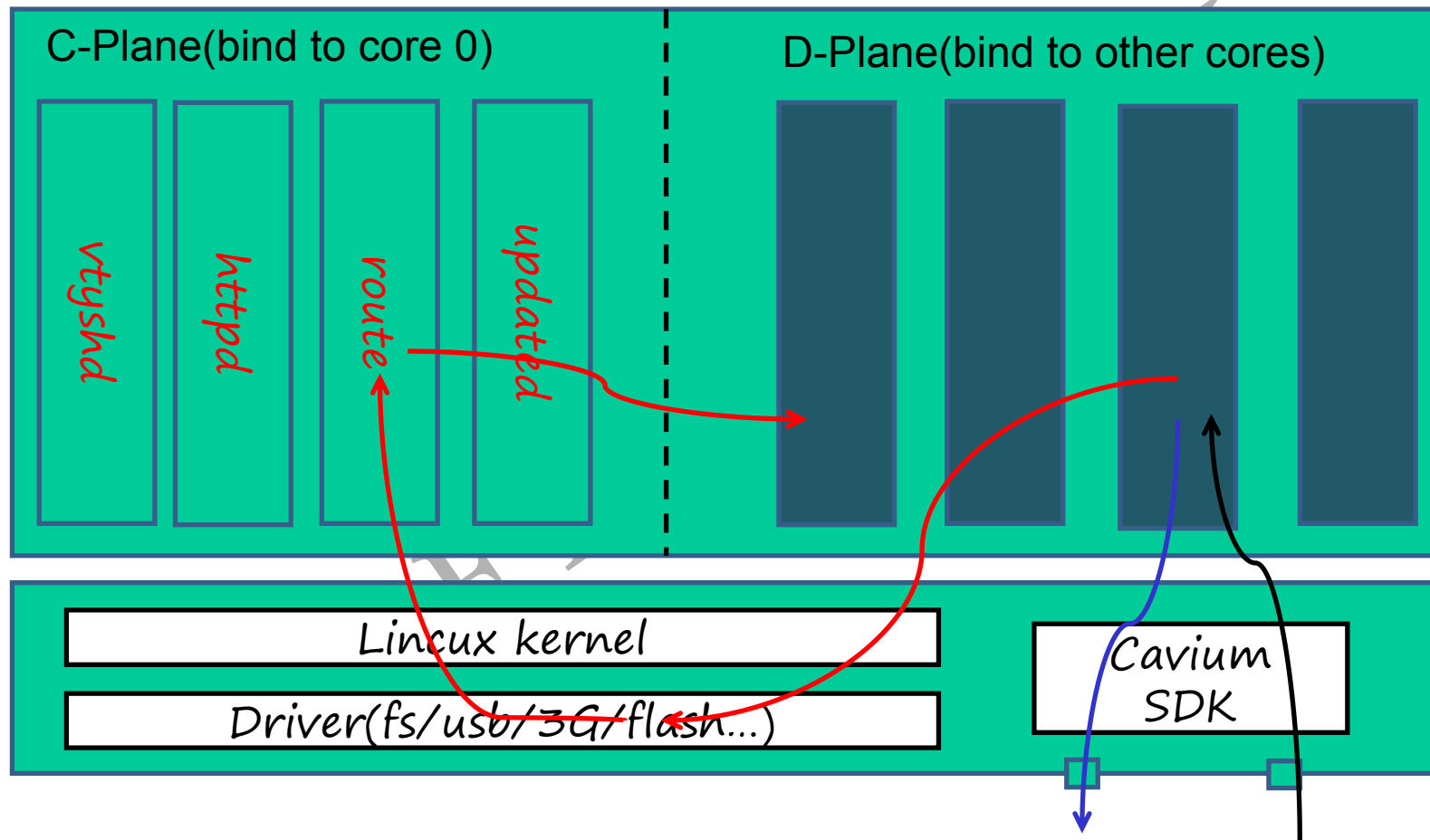
数据共享: 内存映射+自定义内存管理器

进程间通信 ---- TIPC

Lock ---- rcu/rw_lock/spin_lock

Timer ---- 异步机制 (处理流的老化)

C-Plane & D-Plane



协议报文 数据报文

Policy & Session

IPv4策略

☒ 允许
 ☐ 拒绝
 ☐ IPSec

	状态	ID	行为	源	老化时间	日志	老化时间	操作
1	<input checked="" type="checkbox"/>	1	<input checked="" type="checkbox"/>	all	0	ays	-	0 <input type="button" value="编辑"/> <input type="button" value="删除"/>
2	<input checked="" type="checkbox"/>	2	<input checked="" type="checkbox"/>	all	启用 <input checked="" type="checkbox"/>	ays	-	0 <input type="button" value="编辑"/> <input type="button" value="删除"/>
3	<input checked="" type="checkbox"/>	4	<input checked="" type="checkbox"/>	all	匹配条件	ays	-	0 <input type="button" value="编辑"/> <input type="button" value="删除"/>

匹配条件

匹配源

源接口/域

用户

目的接口/域

目的地址

匹配应用和服务

应用

服务

匹配时间

时间

Session信息: 按照五元组进行组织, 用于记录流信息 (识别状态/用户状态等)

两个方向: flow0/flow1

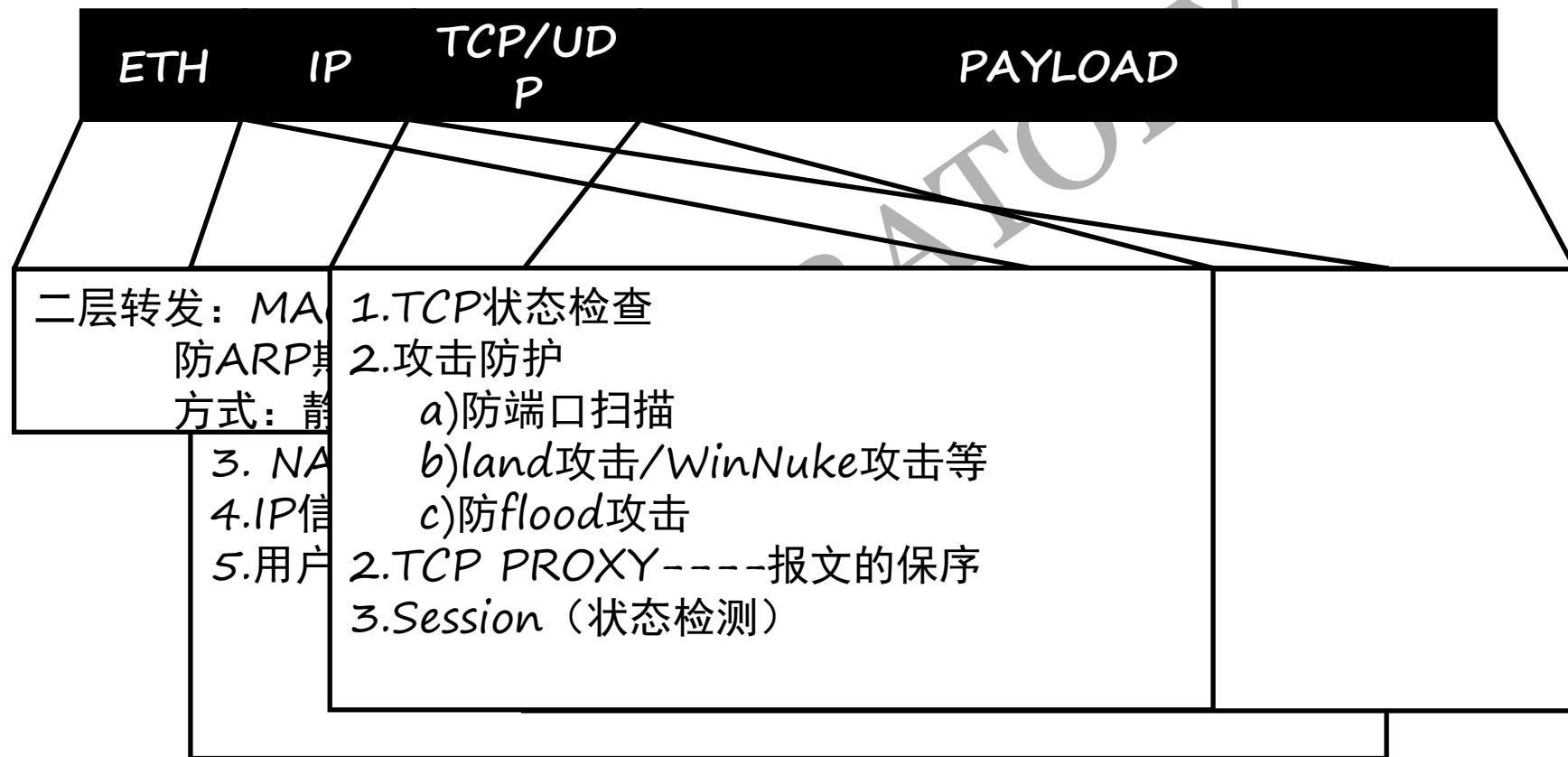
Protocol:TCP State:Complete PolicyID:2

UserName: 172.18.62.180 AppName:QQ

Source Dir: 172.18.62.180:60515 > 121.14.125.46:8080

Reply Dir: 121.14.125.46:8080 > 114.102.65.171:60515

转发流程



应用识别原理与IPS技术

应用识别的作用：

内网安全感知（应用威胁等级）----迅雷/QVOD等都曾爆出漏洞

Policy/策略路由（引流）/QoS

关键技术：

DPI与DFI

流关联技术(FTP/SIP等动态协商协议)

Cache跟踪

DNS解析

Port ---- 知名端口

DPI技术

1.decoder协议解析

IP/TCP/HTTP/FTP/SMTP/POP3/DNS.....

文件重组/ALG(FTP/VoIP/SIP等)

2.signature特征库

Payload : regex or keywords

协议字段组合

支持偏移

3.scan特征匹配

特征提取

Stream Content

```
GET /youku/69712A4862C4D8455D77966B09/03000201004E64B42C4754003E8803078FEB68-  
C5A8-4614-23E6-2EAC36C92011.flv HTTP/1.1  
Host: 118.228.18.33  
Connection: keep-alive  
Referer: http://static.youku.com/v1.0.0187/v/swf/player.swf  
User-Agent: Mozilla/5.0 (windows NT 5.1) AppleWebKit/535.1 (KHTML, like Gecko)  
Chrome/13.0.782.112 Safari/535.1  
Accept: */*  
Accept-Encoding: gzip, deflate, sdch  
Accept-Language: zh-CN, zh; q=0.8  
Accept-Charset: GBK, utf-8; q=0.7, *; q=0.3  
  
HTTP/1.1 200 OK  
Content-Type: video/x-flv  
Accept-Ranges: bytes  
ETag: "941550744"  
Last-Modified: Mon, 05 Sep 2011 11:38:36 GMT  
Content-Length: 1105980  
Connection: close  
Date: Tue, 20 Sep 2011 07:17:35 GMT  
Server: YOUKU.WH  
  
FLV.....  
onMetaData.....metadatacreator..!modified by youku.com in
```

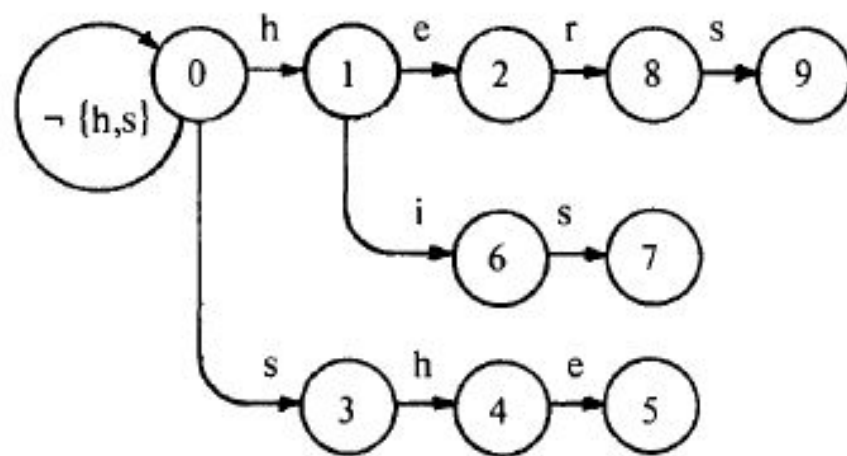
appname: YouKu
regex: GET /youku/.swf/player.swf

快速匹配原理

采用DFA状态机达到O(1)

以AC (多模式匹配算法) 为例说明DFA状态机如何并行扫描

给定模式集{he, she, his, hers}, 其树型有限自动机如下图:



数据结构:

```
typedef struct {
    int NextState[256];
    int id;
}ACSM_STATETABLE;
```

```
p[0].NextState['h'] = 1;
p[1].NextState['e'] = 2;
```

```
p[9].id = appid;
```

<i>i</i>	1	2	3	4	5	6	7	8	9
<i>f(i)</i>	0	0	0	1	2	0	3	0	3

```
USHORT acsmSearch (ACSM_STRUCT * acsm, const UCHAR *Tx, INT n)
{
    INT state;
    const UCHAR *Tend, *Tc;
    USHORT appid = 0;
    ACSM_STATETABLE * StateTable = acsm->acsmStateTable;
    Tend = Tx + n;
    Tc = Tx;

    for (state = 0; Tx < Tend; Tx++)
    {
        state = StateTable[state].NextState[*Tx];
        if( StateTable[state].id != 0 )
        {
            appid = StateTable[state].matchid[0];
        }
    }
    return appid;
}
```


DFA带来的问题

多个正则表达式转换为DFA的问题：

1.merge后状态点膨胀问题(. * {3,5}等)

2.数据结构的优化

稀疏矩阵的压缩(snort)

字符映射(lexertl)

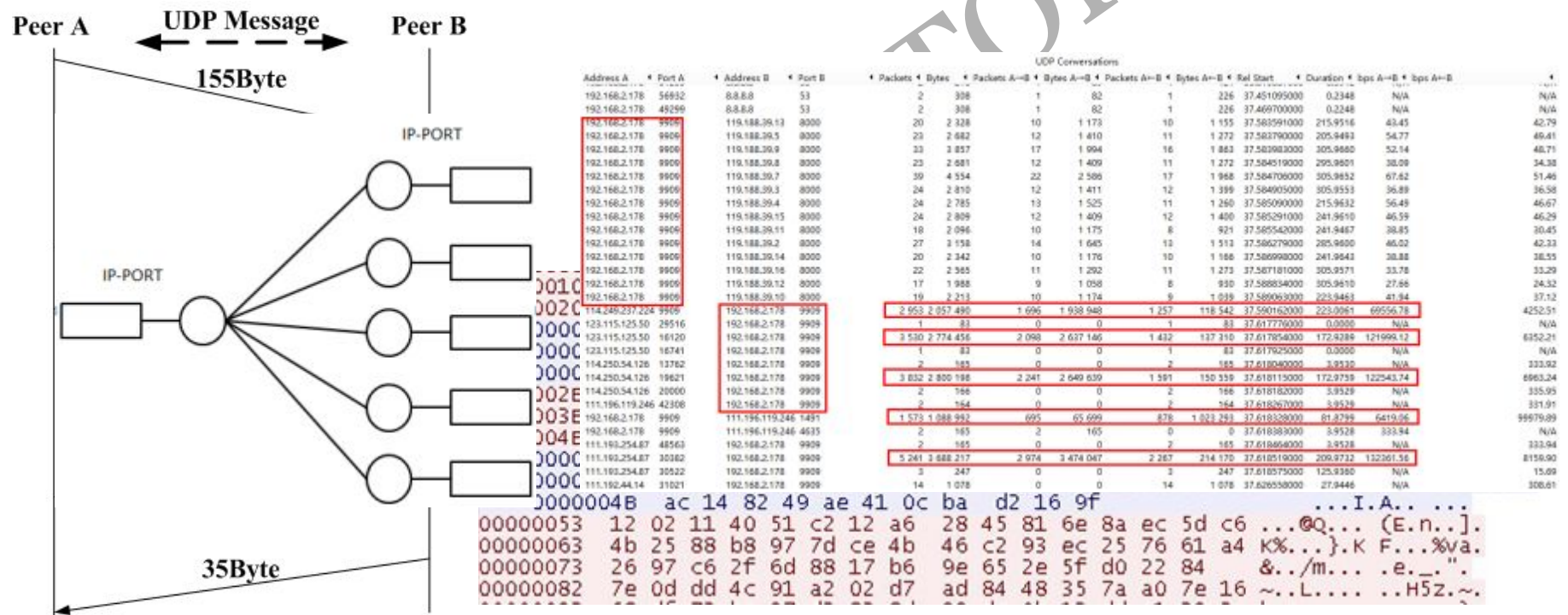
数组索引 vs Trie树

Ushort next[256] vs Void *next[256] (64-Bit)

DFI技术

同一条流内部，交互报文序列长度

- P2P应用流量模型



IPS引擎

不同于应用识别，IPS：

1.对误识别容忍极低

2.匹配条件严苛：

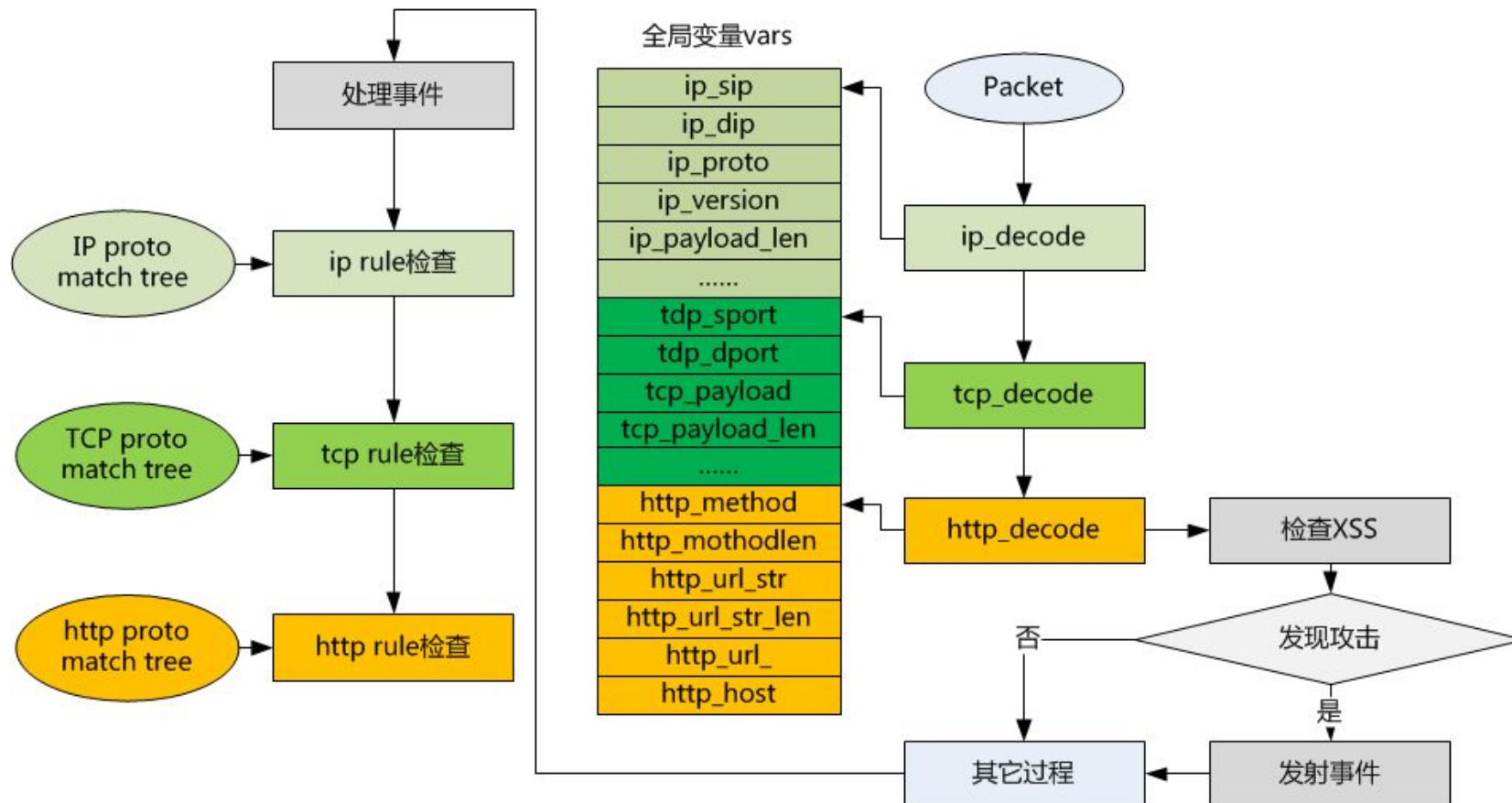
Offset的支持

不同协议变量进行组合

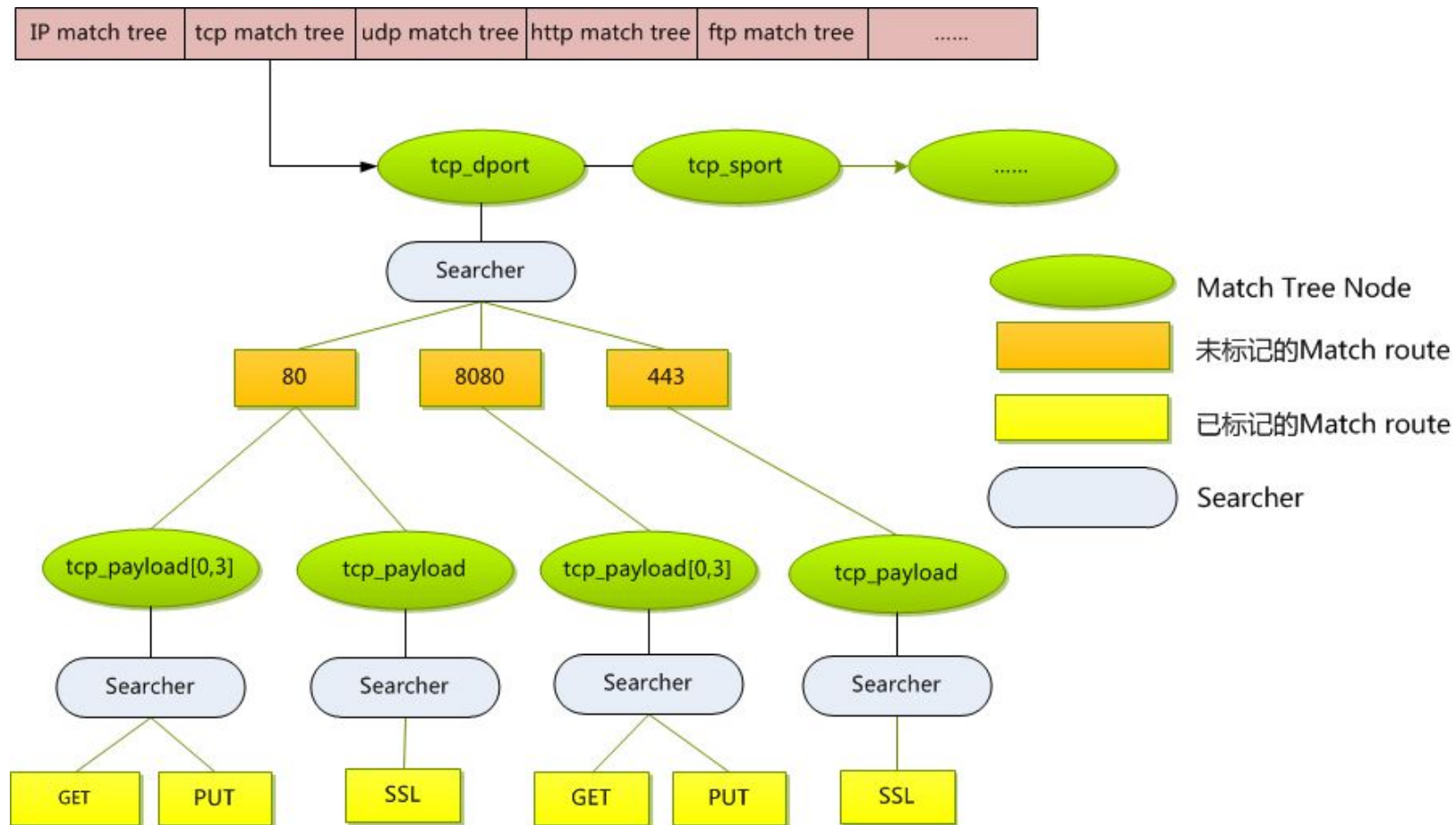
3CDaemon 2.0 FTP Username Overflow (CVE: [2005-0277](#))

tcp_dport=21&tcp_payload[0,4]=USER&tcp_payload_len>245

IPS处理流程



Match Tree ---- abstract search interface





优化思路：

特征顺序调整，尽可能合并相同前缀

采用编译转换技术，将特征转换为等价C语言，ips库变为.so文件

IDF LABORATORY

浅谈防火墙在APT防御中的作用

设置C&C服务器黑名单（黑名单）

网络流量分析（应用识别与IPS技术）

恶意URL检测

高层协议解析与文件还原（av运用的技术）

文件发送到虚拟机中执行！

IDF LABORATORY

未来展望

自身演进与完善

更高的性能、更快的响应速度

完善的特征库

易于部署和维护

统一的策略

用户信息/全网状态可视化

日志的组织和呈现

虚拟防火墙

云端防护

关注我们

- **IDF官网/论坛**

<http://www.idf.cn>

<http://bbs.idf.cn>

- **邮箱联系**

idf@idf.cn

- **关注微博**

新浪微博：@IDF实验室

腾讯微博：@NeteasyIDF

- **黑客文化沙龙QQ群**

204267310

