打造互联网企业全能型WAF

分享人：Drizzle

Part 0x1

需求在哪里？

真命题 OR 假命题

1

我们所处的环境..

安全基础环境/设施还算健全

运维自动化较高 & 研发体系较规范化

安全体系/制度/流程能碰地

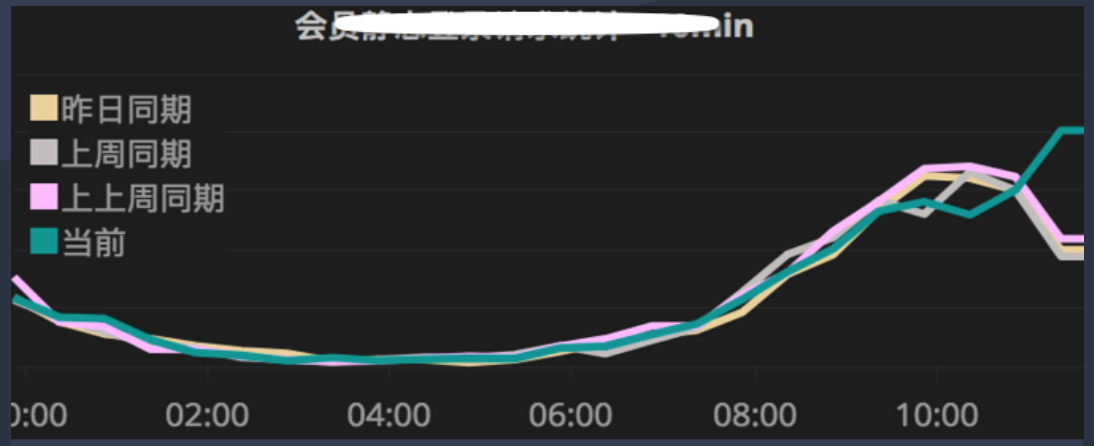
我们面临的问题..

83%

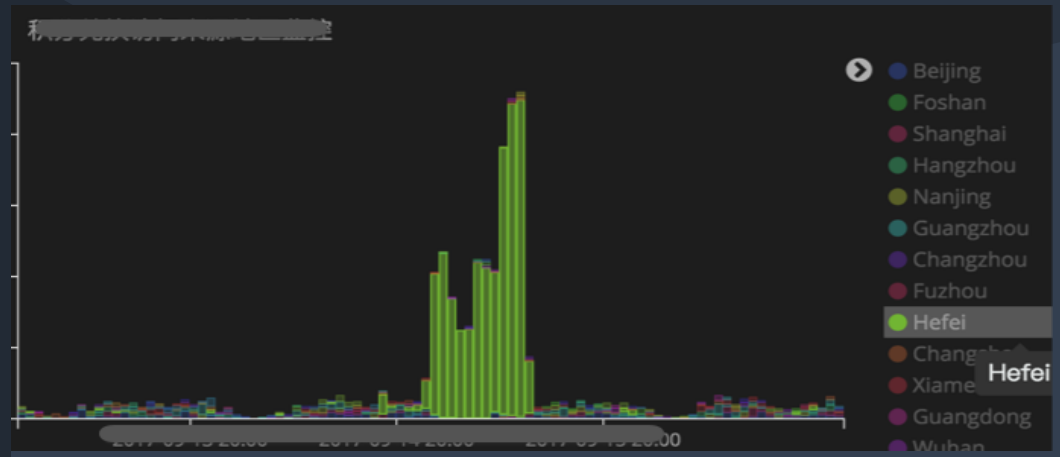
统计来自最近一年，经过安全团队确认，
不同程度影响到业务的十余件安全事件，
83%都可归结于应用安全、业务安全方面

实际看到的攻击

安全监控系统试图打通应用、业务、网络、系统等不同维度的安全相关实时日志，进行关联分析，对攻击进行实时预警，而我们在其中，看到最多的，正是应用或业务层面的安全攻击。



撞库攻击在安全监控系统中的体现



积分盗刷在安全监控系统中的体现

传统的解决方案&现状

如果某业务部门（需求方）遇到了爬虫和羊毛党的威胁...

风控团队

设备指纹系统

反爬系统

反爬团队

野生验证码系统

神秘的封IP规则

风控系统

传说中的业务需求方

好多个黑IP库

WAF系统

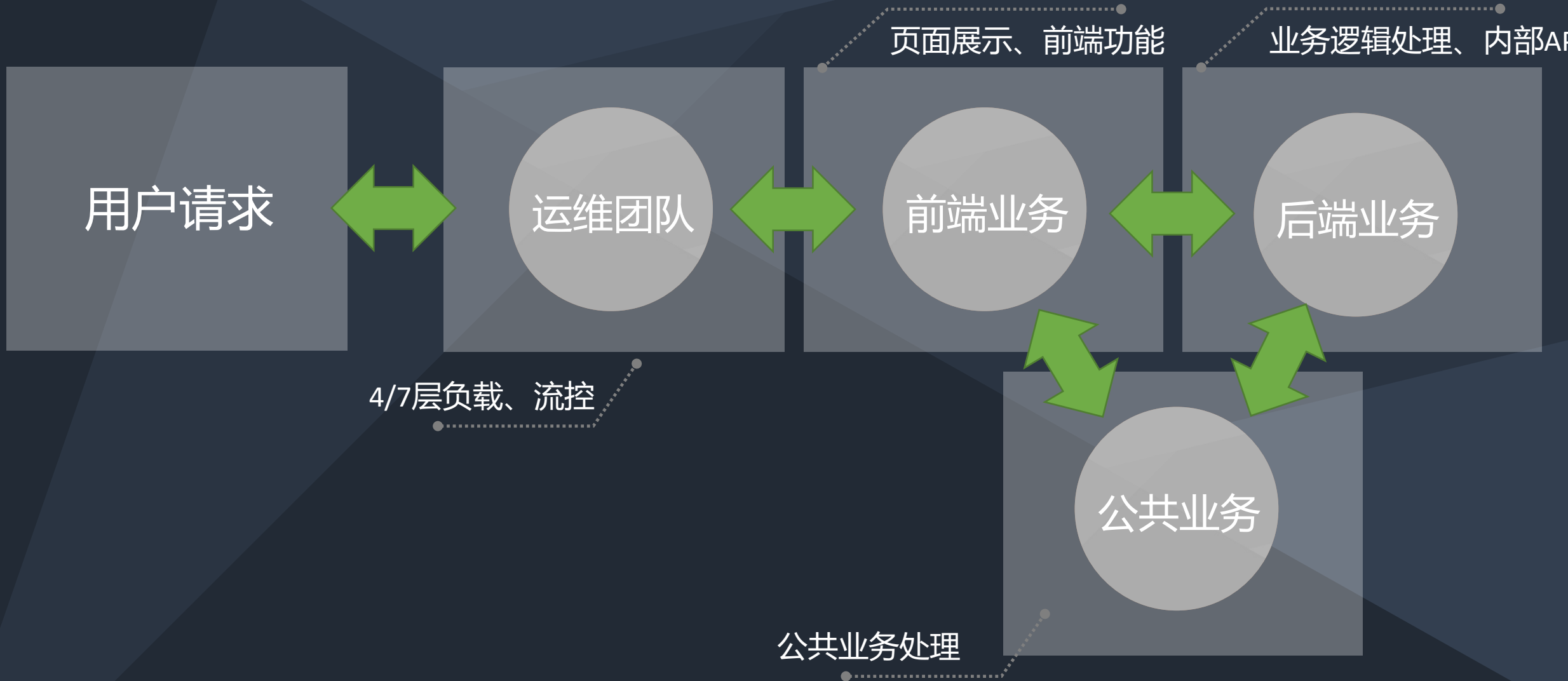
安全团队

安全预警系统

应用限流系统

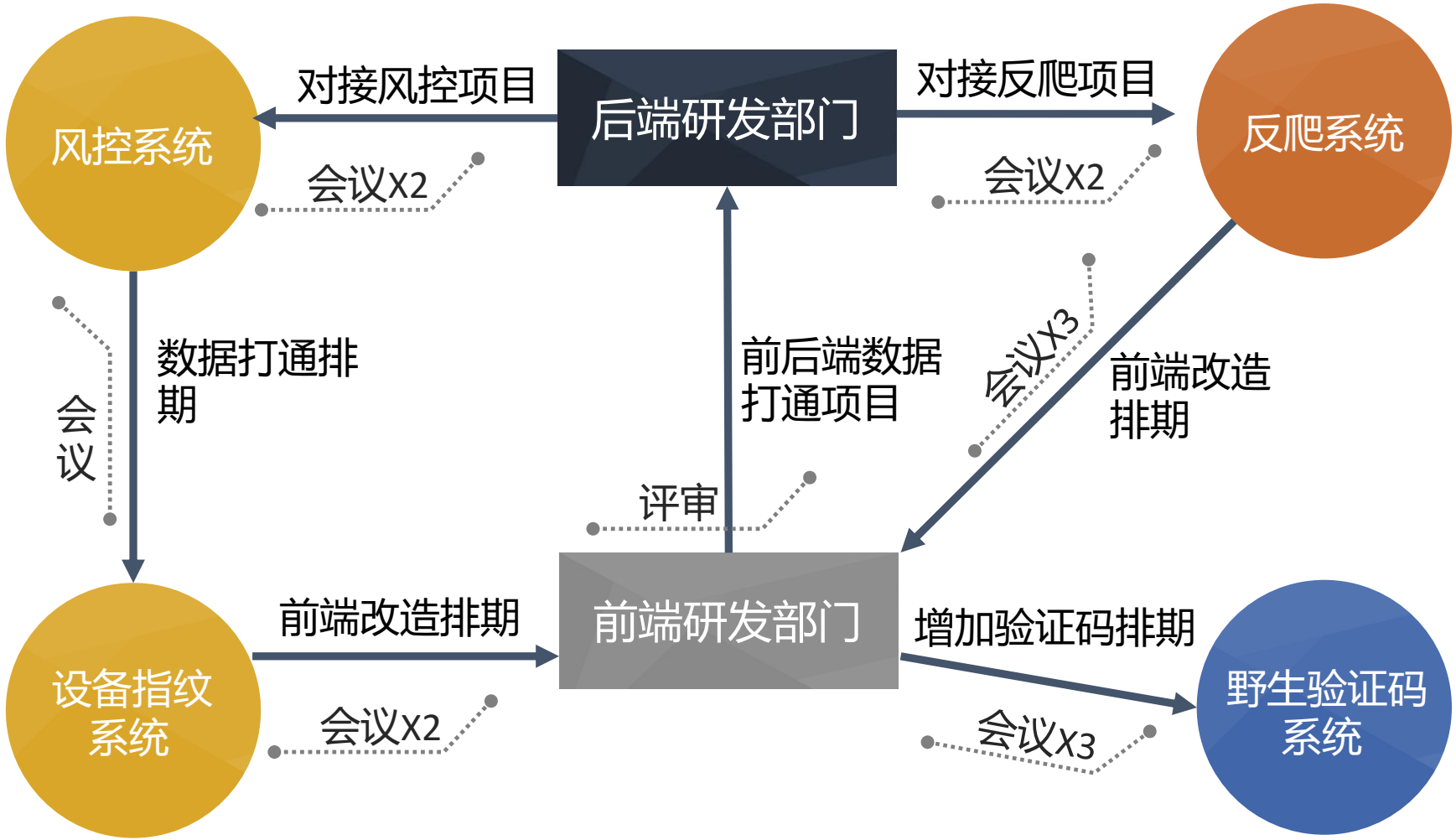
运维团队

请求流向管辖势力范围简图



传统的解决方案&现状

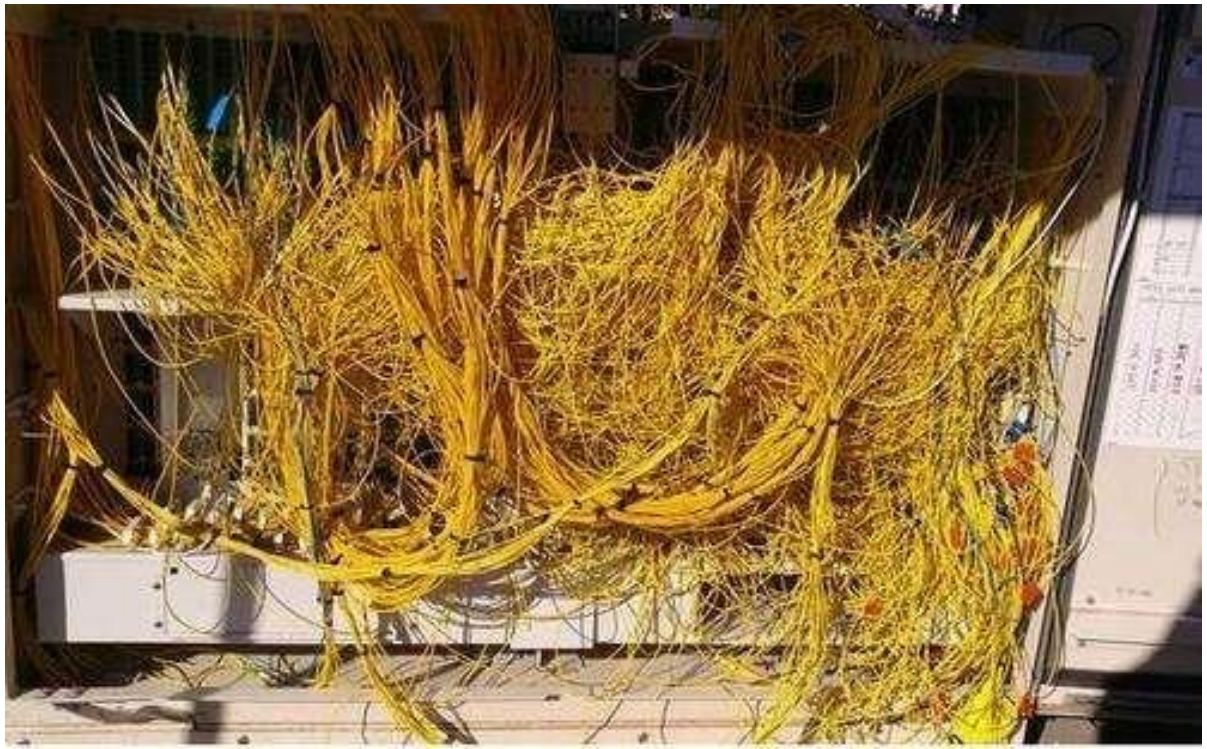
大家都很积极的参与...



涉及5~6个研发团队，无数会议，N次排期，耗费大量资源，2个月过去了...

传统的解决方案&现状

结果经常是...



传说中的业务需求方

各团队研发同学

Part 0x2

为什么不能简单一点？

简单是终极的复杂

我们为什么选择WAF作为切入点

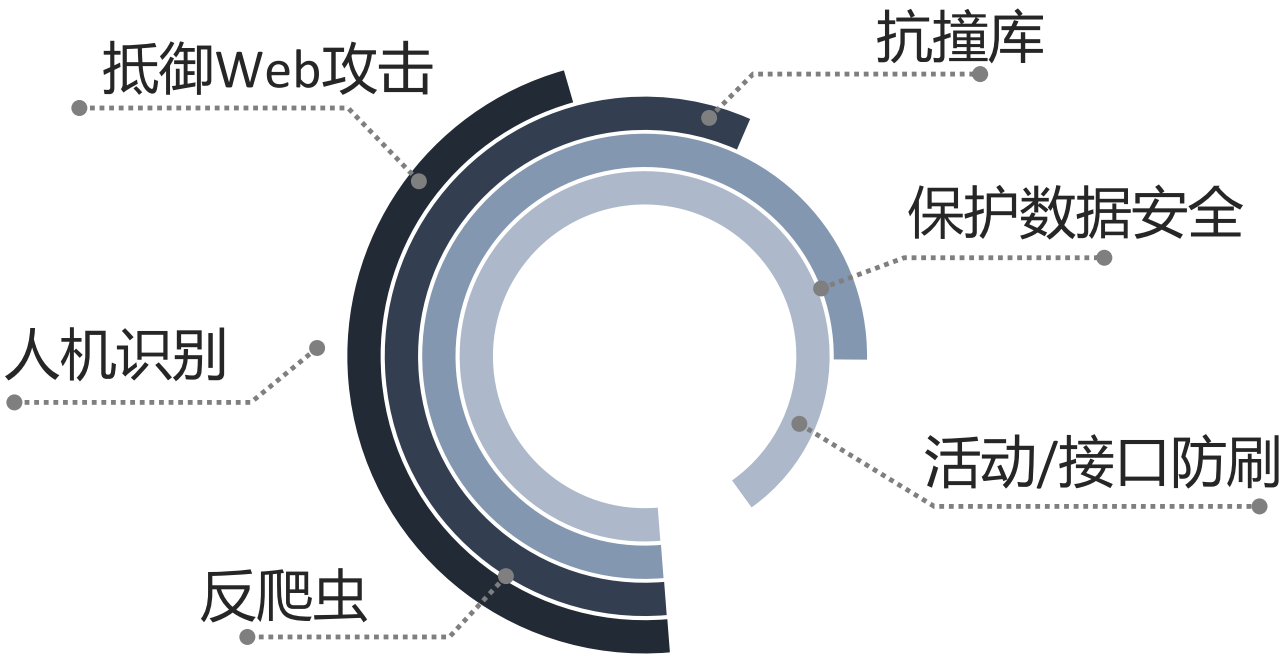
流量入口

WAF身为7层防护系统，处在应用和业务接入的重要位置，卡位很好，对应用和业务有天然亲和力

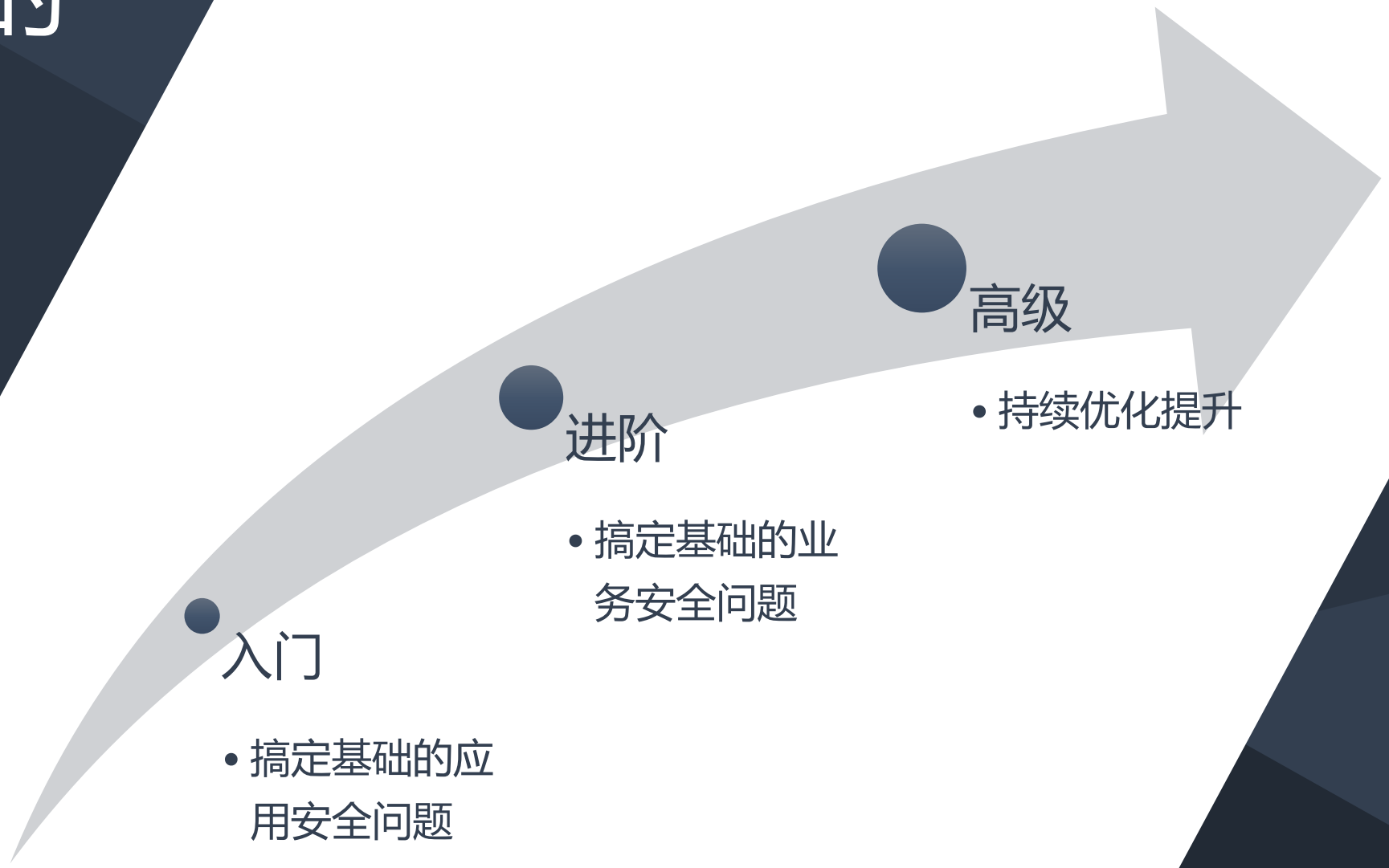
技术场景

互联网应用接入层的流行技术栈：Nginx/Tengine/Openresty的广泛运用，可以快速构建分布式架构，配合lua的灵活性+luajit引擎的高效性，可以构造强大的接入层应用程序

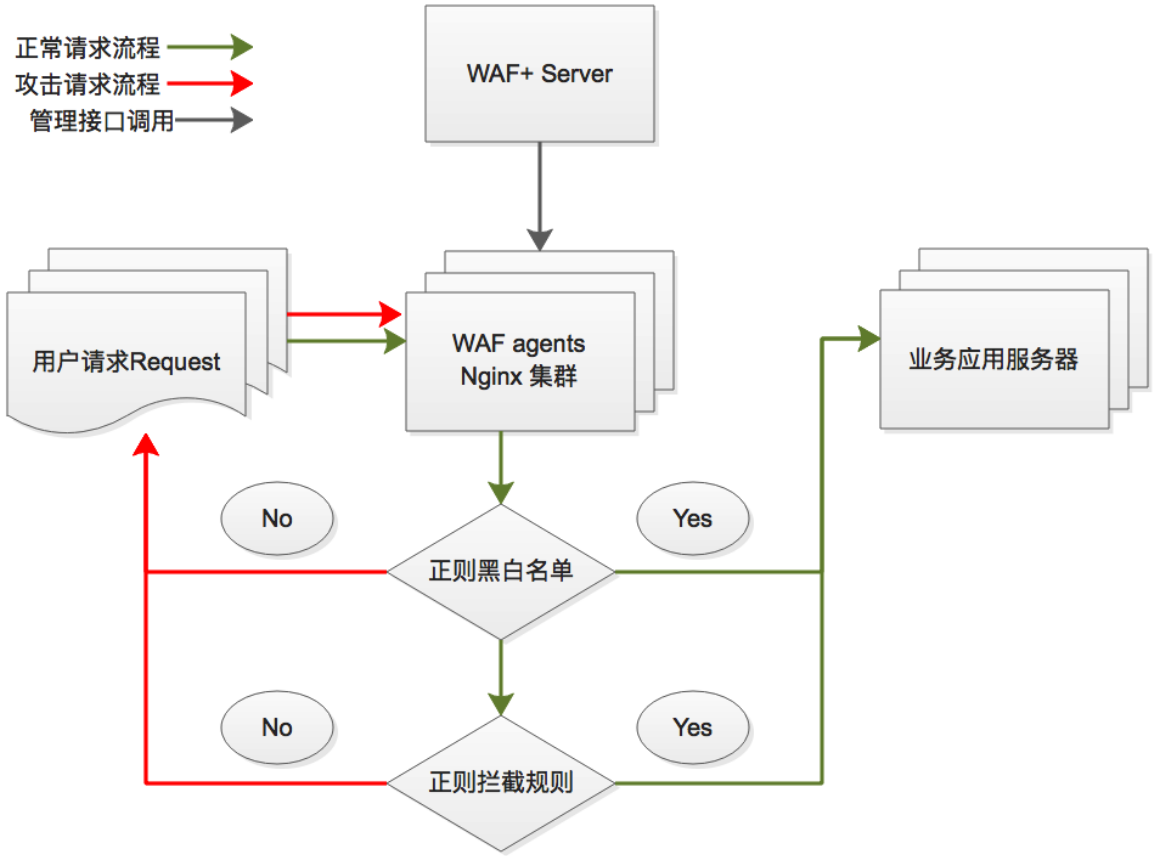
所以，能不能打造一款，能够快速解决大部分应用+业务安全问题的全能型WAF？



我们的思路



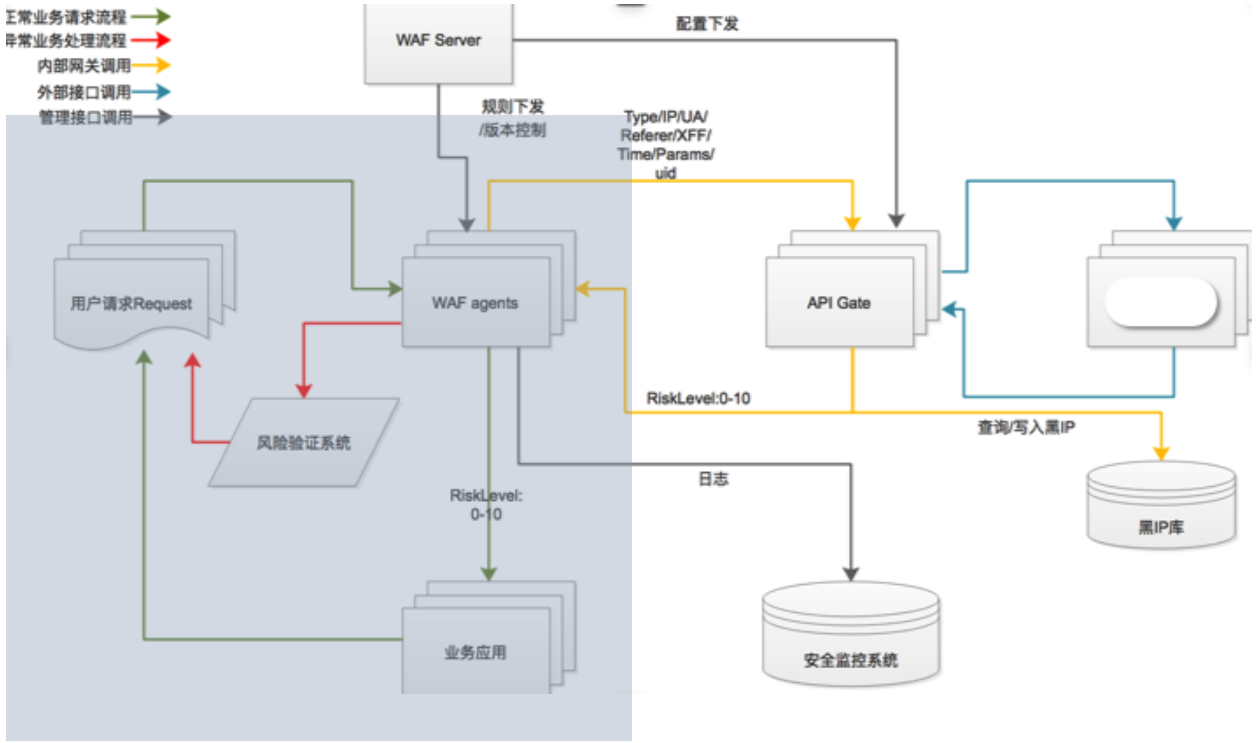
入门



构建基于Openresty的分布式WAF

1、利用Nginx集群反向代理，部署Openresty环境
2、使用lua/luajit编写核心引擎 WAF Agent，进行HTTP包处理
3、Agent使用正则进行拦截规则和名单匹配
4、使用Python构建 WAF Server，进行规则、名单管理/配置下发
5、采集WAF日志到ELK进行分析展示

进阶



集成&对接业务安全模块

在应用接入层保护业务安全

1、通过集成，在应用接入层接入风控系统
2、通过埋JS和集成，在应用接入层接入设备指纹系统
3、通过埋JS和集成，在应用接入层接入风险验证系统

技术点exp

风险验证系统（模块）：

1.利用Openresty在响应包中注入定制JS

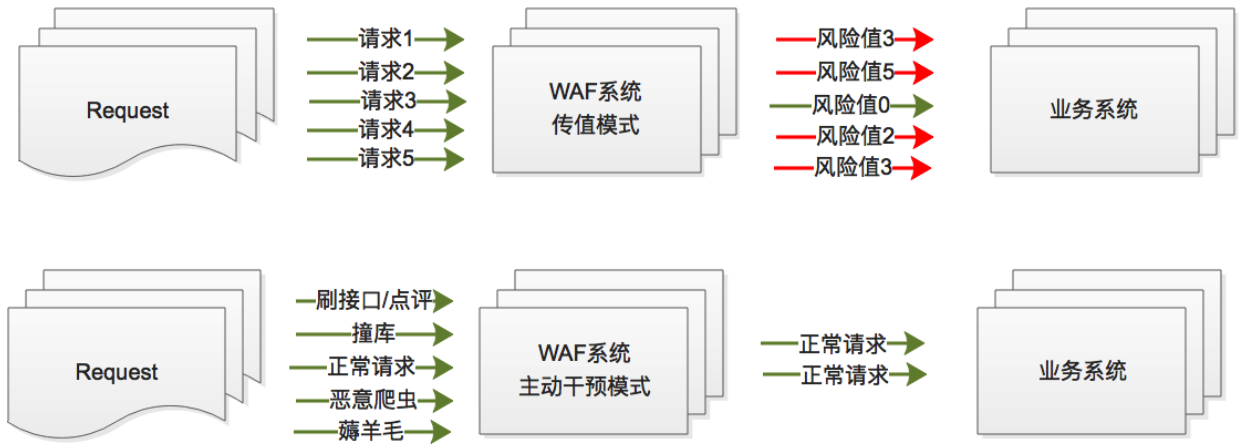
2.JS Hook Ajax/JSONP/Href...

3.JS搜集设备环境信息

4.制作完美兼容各种浏览器的浮层

5.集成一套完善的人机识别系统（如行为验证码）

业务视角



传值模式&主动干预模式

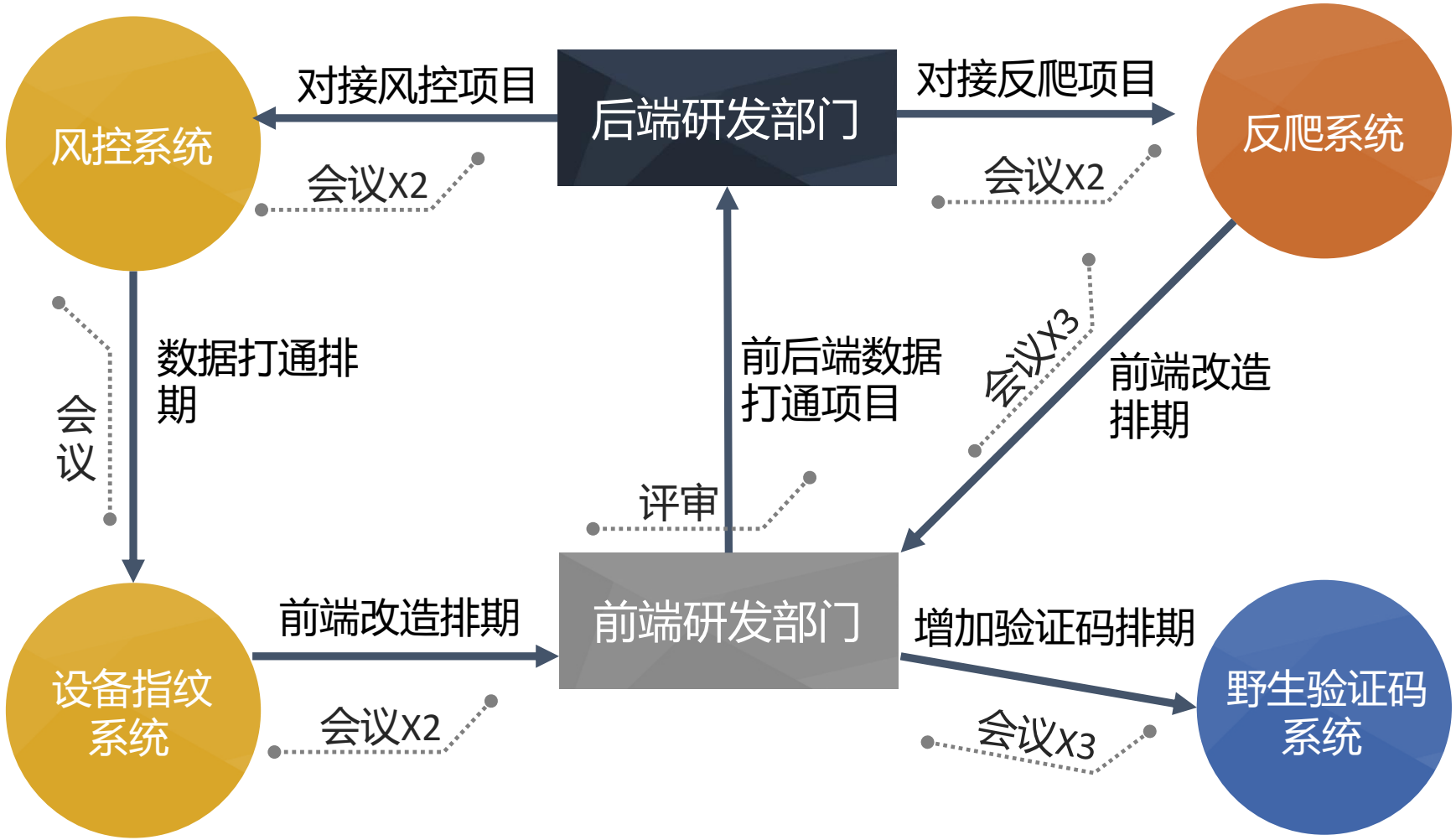
在应用接入层保护业务安全

1、传值模式：通过WAF集成的多种模块，根据不同的场景，计算每一个请求的风险值，通过HTTP附加字段，传给业务系统，由业务系统自主决定如何处置

2、主动干预模式：计算出风险值后，根据风险值的区间，利用风险验证系统，选择合适的人机识别方式，进行主动干预，请求由WAF接管，未通过风险验证不能到达业务系统

传统的解决方案&现状

回顾一下以前的方式...



涉及5~6个研发团队，无数会议，N次排期，耗费大量资源，2个月过去了...

新的业务安全方案

现在多了一种选择...

简单沟通，页面操作配置，灰度测试，上线... 问题解决

Part 0x3

畅想高级玩法

想象是第一生产力

WAF引擎优化

基础应用防护引擎，抛弃正则，借助机器学习、语义分析进行优化

WAF功能优化

借助WAF的优势横向扩展功能，如UGC自动过滤、敏感数据自动打码，将WAF打造成应用和业务安全的基础设施

THANK YOU!