宜信安全应急响应中心
CreditEase Security Response Center

FREETALK

2018 北京站

可扩展的分布式安全扫描与响应平台

FREEBUF

公司介绍：

- 宜信——创立于2006年，从事普惠金融和财富管理事业的金融科技企业。在支付、网贷、众筹、机器人投顾、智能保险、区块链等前沿领域积极布局，通过业务孵化和产业投资参与全球金融科技创新。
- 成立十二年以来，始终坚持以理念创新、模式创新和技术创新服务中国高成长性人群、大众富裕阶层和高净值人士，真正的让金融更美好。

部门介绍：

- 宜信安全部直属于宜信总公司，承接总部及所有子公司的安全需求，为公司的业务安全保驾护航。

  ※安全部成4年以来，弥补了公司安全基础建设的缺失，发展至今组建了覆盖网络、应用、主机、终端、数据逐层渗透的前沿安全架构，每一步都凝聚着肩负亿万用户信息和财产安全的责任。

  ※ 2018年我们在传统安全架构上引入了当前热点科技，将人工智能与大数据技术融合到当前的安全架构中，将资产、漏洞、风控和管理有机结合到一起，构建用户画像，为实现安全态势感知与威胁情报预警奠定了基础。

  ※ 自研的【洞察】漏洞管理平台在GitHub上进行了开源，借此帮助更多金融科技的行业伙伴，完成自动化风险全生命周期的管理并实现风险的可量化，共建互联网金融行业的安全生态。

问题和痛点

安全产品学习和运维成本高

- 产品、工具种类繁多，学习和切换成本较高
- 人工使用工具获取资产数据和进行结果处理耗时耗力
- 安全工具功能单一，且无法协同使用

安全检测网络环境多样化

- IDC（多IDC ）,办公网，公网……单一扫描节点无法满足复杂的需求

解决思路

疲于工具的学习和使用

乐于技能提高，专注风险本身

0x01 谈谈分布式

基于任务队列的分布式扫描

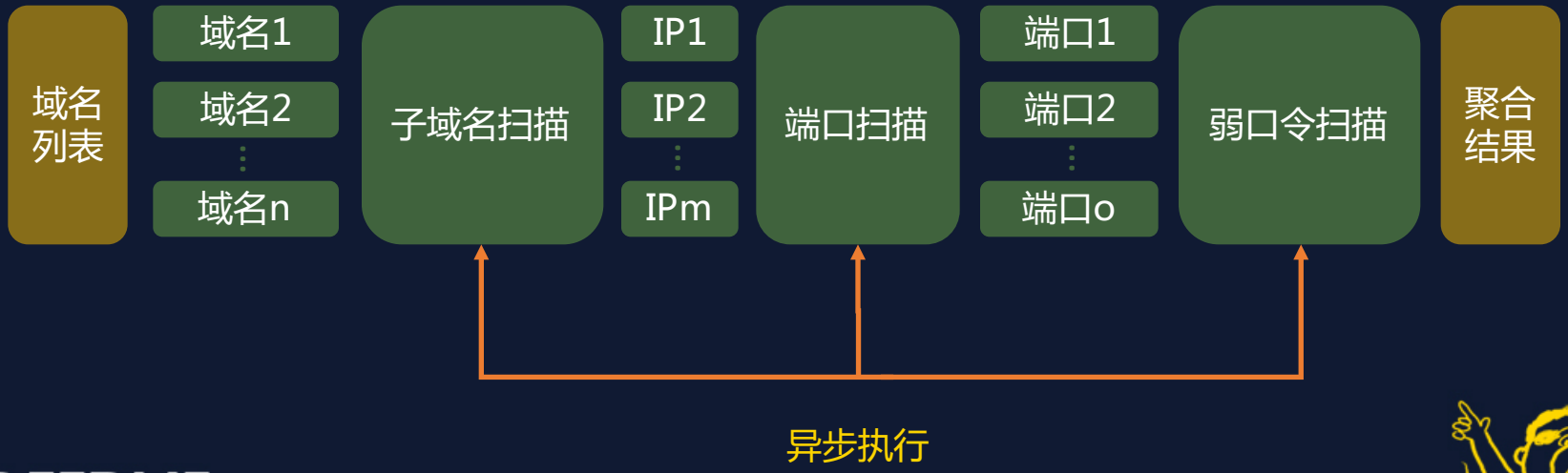
1. 难以将扫描功能解耦合
2. 分发后聚合实现比较繁琐
3. 扫描模块调用接口不统一

基于MapReduce模型分布式扫描

1. 易于任务数据分片和结果聚合
2. 调用链使用简单
3. P2P方式进行数据传输，效率高

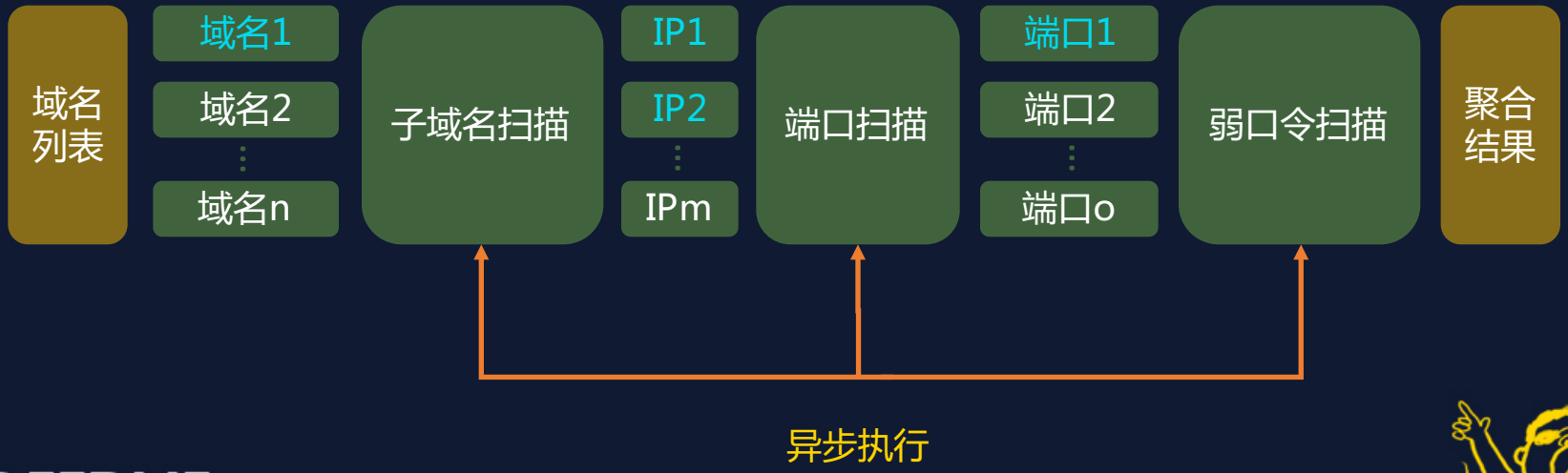
举个栗子

- 举例说明任务数据分片和任务链过程: 子域名扫描->端口扫描->弱口令扫描



举个栗子

- 举例说明任务数据分片和任务链过程: 子域名扫描->端口扫描->弱口令扫描



0x02 功能可扩展（易于开发）

插件开发

- 开发过程简化
- 将重复的工作抽象出来封装起来，例如安全通信，任务监控，数据传输等等
- 提供协程，线程，进程任务执行粒度支持
- Python依赖自动检测并安装
- 结果保存为JSON方便后续使用

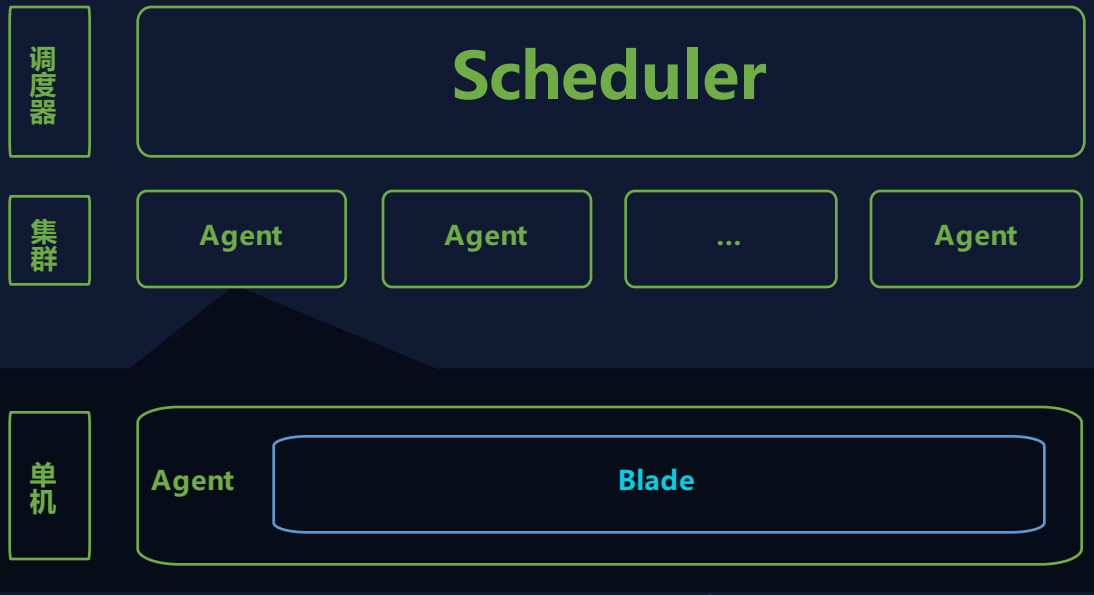
HelloWorld示例

```
class TestPlugin(Plugin):
    '''
    测试插件
    '''

    def operate(self, **param):
        '''执行操作'''

        url_list = param['url_list']

        for url in url_list:
            html = urllib.urlopen(url).read()
            m_title = re.search(r'<title>(.*)</title>', html, flags=re.I)
            title = m_title and m_title.group(1) or 'UNKNOWN'
            self.logger.info('URL: %s , TITLE: %s' % (url,title))
            self.push({'DATA': {'TITLE': title, 'URL': url}})
```


任务状态

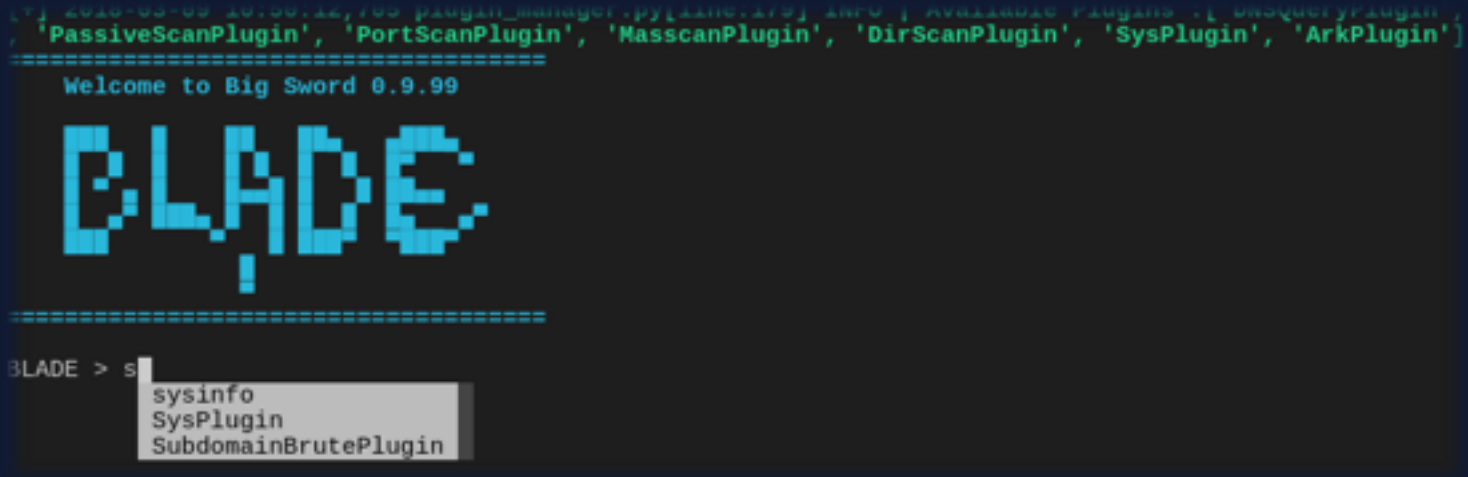
CRON表达式，任务参数，执行次数，执行花销时间，优先级等信息

Blade扫描器与分布式平台



便于单机开发和测试

Blade扫描器-命令行模式



宜信安全应急响应中心
CreditEase Security Response Center

FREETALK
2018 北京站

Blade扫描器-Web模式

FREEBUF

插件易于分发

- 复用，一键打成Zip包和Zip包导入
- vs PoC 偏向于功能级别的插件

多种响应方式

- 平台响应
- 邮件响应
- 第三方即时通讯接口响应

平台响应

漏洞名称，描述以及扫描和更新时间

邮箱响应

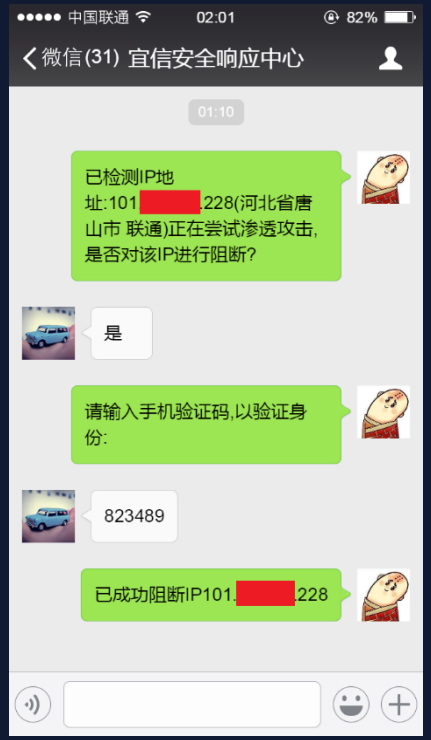
任务运行时的异常

计划任务执行异常 AWVSPlugin

件人

{
'TASK_ID': u'HmpgZi7nZBFj+vKcnbb8hnujXYfYqwGg36U4iBIW9/s=',
'ERROR': 'Traceback (most recent call last):\n File "../cesec_host_agent/core/plugin.py", line 136, in init_thread\n self.operate(**param)\n File "/app/agent/cesec_host_agent/plugins/AWVSPlugin.py", line 129, in operate\n if wvs.is_finished(item):\n File "/app/agent/cesec_host_agent/plugins/AWVSPlugin.py", line 354, in is_finished\n req = requests.post(url, headers=self.headers, data=json.dumps(data), verify=False)\n File "/usr/lib/python2.7/site-packages/requests/api.py", line 112, in post\n return request(\'post\', url, data=data, json=json, **kwargs)\n File "/usr/lib/python2.7/site-packages/requests/api.py", line 58, in request\n return session.request(method=method, url=url, **kwargs)\n File "/usr/lib/python2.7/site-packages/requests/sessions.py", line 502, in request\n resp = self.send(prep, **send_kwargs)\n File "/usr/lib/python2.7/site-packages/requests/sessions.py", line 612, in send\n r = adapter.send(request, **kwargs)\n File "/usr/lib/python2.7/site-packages/requests/adapters.py", line 504, in send\n raise ConnectionError(e, request=request)\nConnectionError: HTTPSConnectionPool(host=\'10.143.128.77\', port=9999): Max retries exceeded with url: /api/getScanHistory (Caused by NewConnectionError(\': Failed to establish a new connection: [Errno 111] Connection refused\',))\n',
'RUNNING_STATUS': 'FAILURE',
'LAST_START_TIME': 1524128960.601824,
'LAST_FINISH_TIME': 1524260380.443133,
'DATA': [], 'OPERATION': False,
'PLUGIN_NAME': 'AWVSPlugin',
'LAST_TIME': 131419.8413090706}

第三方即时通信响应


中国联通
02:01
82%
微信(31) 宜信安全响应中心
01:10
已检测IP地址:101[redacted].228(河北省唐山市 联通)正在尝试渗透攻击,是否对该IP进行阻断?
是
请输入手机验证码,以验证身份:
823489
已成功阻断IP101.[redacted].228

0x03 架构可扩展

分布式架构


Web 层
Ngnix
Gunicorn
前后端分离
JWT认证和鉴权
Web
MSG Consumer
DB
Flask APP
Flask APP
Flask APP
Flask APP
调度层
Scheduler
MQ - Redis
负载均衡
Python原生实现定时任务
SQLite
通信层
JSON/TCP
ZMQ+IOLOOP 异步高并发
授权通信
超时重传
压缩与序列化
集群
Agent
Agent
...
Agent

架构扩展


Ngnix
Load Balancing
Gunicorn
Gunicorn
Gunicorn
Web
Consumer
Web
Consumer
Web
Consumer
Flask APP
Flask APP
Flask APP
Flask APP
Flask APP
Flask APP
DB Proxy
DB Write
DB Read
DB Read
Broker
Scheduler
Scheduler
Scheduler
Redis Cluster
Agent
Agent
Agent
Agent
Agent
Agent

0x04 节点可扩展

节点简单部署

- 1. 一键安装 pip install blade.whl
- 2. 配置Scheduler Server地址并启动，如 192.168.1.2:5559

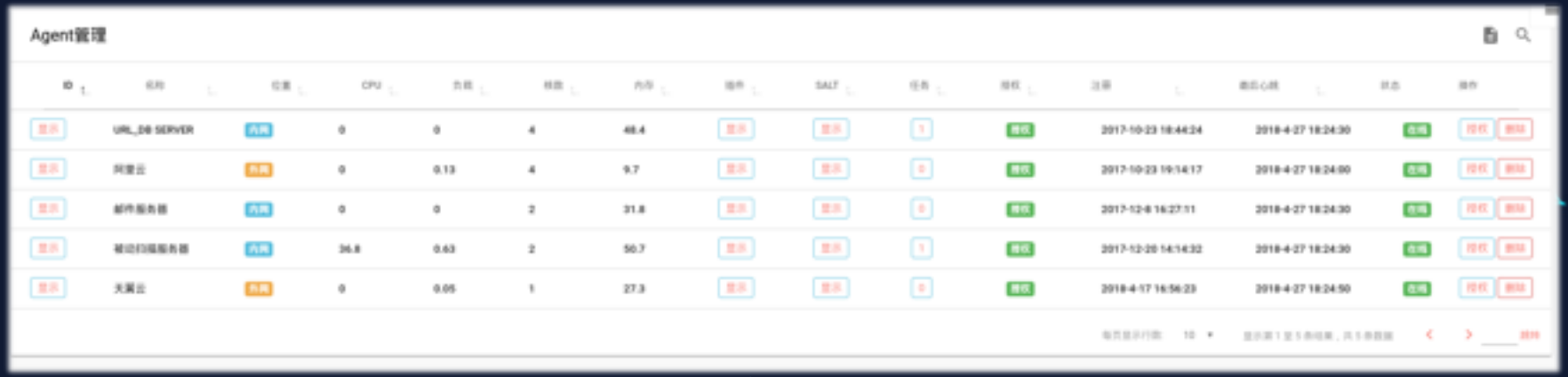
支持配置文件，方便使用自动化运维分发部署

如有环境依赖：提供Docker file生成

节点授权与通信

- TCP+JSON通信，自定义协议
- Scheduler授权，Token认证

节点授权与通信



节点位置，CPU使用率，负载，内存使用率，CPU核数位置等

节点超时重传机制

- 支持节点与调度器之间进行单向通信
- 如果网络故障可保存数据尝试重连
- 超时时间可配置

0x05 展望

展望

- 1. 失效转移
- 2. 阻塞方式的数据转发（分布式同步请求服务）

宜信安全应急响应中心
CreditEase Security Response Center

FREETALK
2018 北京站

Get more information ↓

FREEBUF