

Akamai Security Summit World Tour

撞库、身份和欺诈 加强保护并介绍新策略

Unmesh Deshmukh 阿卡迈亚太区云安全域副总裁



Intelligent Security Starts at the Edge

●●●○○ 我的运营商

12:00

21% 



编辑

大家好！

登录以访问您的资金。

 用户名

 密码

登录

```
package main; import ( "fmt"
"target string; Count int64; }
pool); statusPollChannel := m
select ( case respChan :=
true; go doStuff(msg, resp
chan ControlMessage,
quest) { /* Does an
r.ParseForm();
return; } msg :=
issued for /s
http.ResponseWriter,
time.Second);
}; return
import (
```



```

);
fmt.Fprintln(w, "
");
nil; }
message string;
respChan :=
statusPollChanne
workerAc
status; )); fu
http.ResponseWriter,
strings.Split(r
); nil { fmt.Fprintf(w,
%v\n", msg; fmt.Fprintf(w,
count); }); http.HandleFunc("/s
PollChannel <- reqChan; timeout
"ACTIVE"); } else { fmt.F
```

```
time.After(time.Second); select ( ca
```



编辑



Josh Shaul

@Josh_Shaul
于昨天成为会员

账户余额: **\$2.00**



Josh Shaul 向 YouGotPwned 付款
账单分类: 10QSucka

1 分钟前

赞 评论

-\$1,999.00



Josh Shaul 向 Need Mulaah 付款
账单分类: 酒和药品

2 分钟前

赞 评论

-\$1,500.00



Josh Shaul 向 AdultFriendFinder 付款
账单分类: 交友

3 分钟前

赞 评论

-\$1,000.00



Josh Shaul 向 Joe Smith 付款
账单分类: 有趣的网络内容

4 分钟前

赞 评论

-\$1,999.00



编辑



Josh Shaul

@Josh_Shaul
于昨天成为会员

帐户余额: **\$2.00**



Josh Shaul 向 YouGotPwned、Need Mulaah、AdultFriendFinder 和 Joe Smith 付款

-\$6,498.00

赞

评论

怎么回事?



赞 评论

-\$1,500.00



Josh Shaul 向 AdultFriendFinder 付款
账单分类: 交友

3 分钟前

赞 评论

-\$1,000.00



Josh Shaul 向 Joe Smith 付款
账单分类: 有趣的网络内容

4 分钟前

赞 评论

-\$1,999.00



Josh Shaul

@Josh_Shaul
于昨天成为会员

帐户余额: **\$2.00**



Josh Shaul 向 YouGotPwned、Need Mulaah、AdultFriendFinder 和 Joe Smith 付款

-\$6,498.00

- 赞
- 评论
- 怎么回事?


```
package main; import ("fmt" "net/http";
target string; Count int64; )
pool); statusPollChannel := make(chan
select ( case respChan := <<= 1
true; go doStuff(msg, respChan)
chan ControlMessage; respChan := reqChan
quest) ( /* Does an http request */
r.ParseForm()); count := count + 1
return; ); msg := "hello world"
issued for Target := target; respChan :=
http.ResponseWriter; statusPollChannel :=
time.Second); } return; }
import ("fmt" "net/http";
Count int64; )
PollChannel := make(chan ControlMessage);
case respChan := <<= 1
doStuff(msg, respChan)
pool);
```



我们来谈论一下撞库

```
func (w *Worker) doStuff(msg string, respChan chan
} timeout := 10 * time.Second
fmt.Fprintf(w, "Request: %s\n", req);
"333", nil);
message string; reqChan := reqChan; respChan :=
respChan := respChan; statusPollChannel :=
channel; workerAccount := workerAccount;
status; }); func doStuff(msg string, respChan chan
http.ResponseWriter; statusPollChannel :=
strings.Split(req.RawQuery, "&");
nil ( fmt.Fprintf(w, "Request: %s\n", req);
"333", nil);
msg; fmt.Fprintf(w, "Request: %s\n", req);
count); ); http.HandleFunc("/status");
PollChannel <- reqChan; timeout := 10 *
time.Second); } else { fmt.Fprintf(w, "ACTIVE");
```

欺诈杀链

第 I 部分 - 了解撞库的作用



- 识别帐户价值高的目标网站
- 在暗网上购买被盗帐户信息的列表



- 购买或租用僵尸网络以自动执行验证
- 构建或购买软件工具以逃避检测



- 在目标网站的登录页面上验证被盗帐户的信息列表
- 在暗网上重新销售被验证过的帐户信息

目标

解决方案

爬虫程序管理

- 在杀链早期缓解攻击，以降低下游欺诈的发生率

侦测

武器化

交付

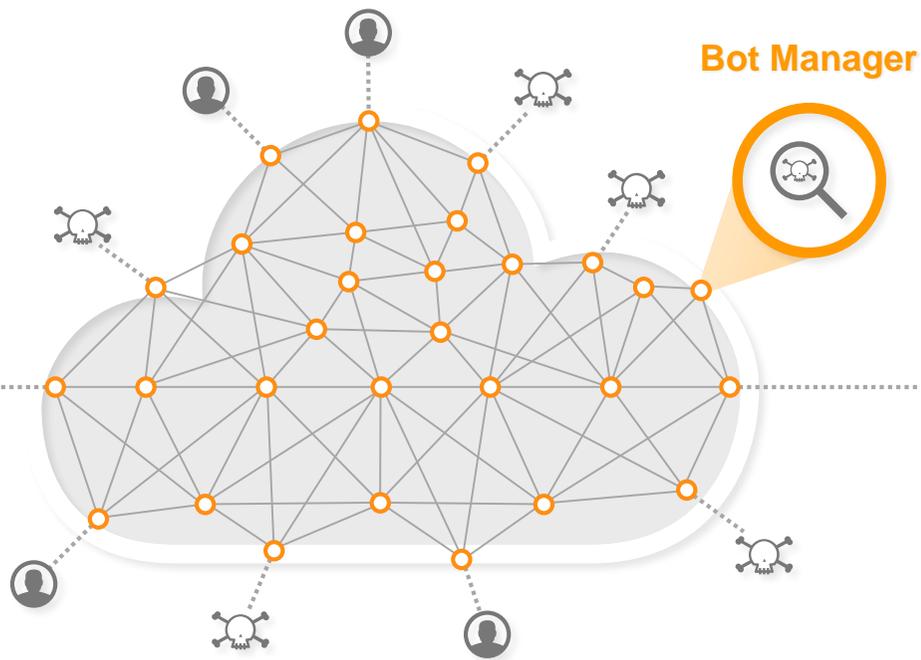
利用

措施

BOT MANAGER PREMIER

撞库属于爬虫程序问题

用户在 Akamai 边缘
进行身份验证



Web 应用程序



注册



登录

- 超过 20 层爬虫程序检测
- 无监督机器学习 + 深度学习技术
- 针对人工流量和爬虫程序流量的出色可见性

高级爬虫程序检测

机器学习由出色的数据访问提供支持

客户端数据

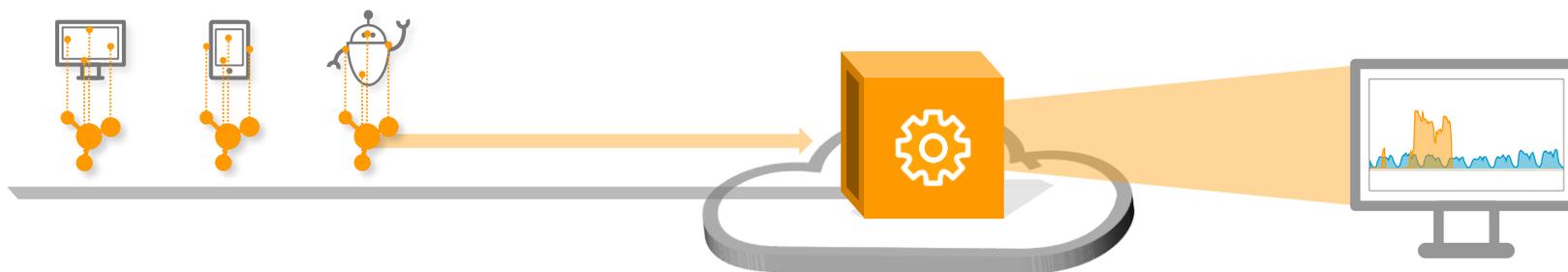
客户端数据收集

分析引擎

异步服务器端分析

爬虫程序检测

人类或更高准确度的爬虫程序



-  用户行为信号
-  设备 + 浏览器特征
-  每天 13 亿台客户端设备

-  数百个信号的信号处理
-  监督学习、无监督学习和深度机器学习模型

-  高准确度；目标 <0.2% FP
-  不受预先构想的爬虫程序定义的约束

案例研究

全球前十大金融服务机构

作为最大的金融资产管理公司之一，该组织看到了大量的爬虫程序流量，包括金融整合程序以及撞库和其他与欺诈有关的活动。



问题

每月 8000 次帐户接管，跨越多个登录端点，导致每天直接欺诈相关损失高达 10 万美元



解决方案

在每个消费者登录端点前以“拒绝”模式部署基于行为的爬虫程序检测



结果

在所有登录端点上，每月的帐户接管数量明显下降到 1-3 个，每天的欺诈相关的损失显著降低到 1-2 千美元

2

身份可被用于哪些非法目的

欺诈杀链

第 2 部分 - 身份如何将我们与欺诈活动联系在一起

目标



- 识别帐户价值高的目标网站
- 在暗网上购买被盗帐户信息的列表



- 购买或租用僵尸网络以自动执行验证
- 构建或购买软件工具以逃避检测



- 在目标网站的登录页面上验证被盗帐户的信息列表
- 在暗网上重新销售被验证过的帐户信息



- 购买已被入侵的目标网站帐户
- 使用所购帐户信息登录



- 使用已被入侵的帐户执行欺诈性交易

解决方案

爬虫程序管理

- 在杀链早期缓解攻击，以降低下游欺诈的发生率

身份

诈骗预防

- 优点：了解个别用户
- 缺点：成本高，帐户已遭遇入侵

侦察

制作攻击工具

交付

攻击

措施

```
package main; import
target string; Count
pool); statusPollCh
select ( case res
true; go doStatus
```

通过收购 Janrain, Akamai 拥有了可增强数字信任的客户身份管理功能



通过减少登录和注册工作负载，Janrain 使企业能够增强数字信任。

了解 CIAM

客户身份和访问管理

什么是 CIAM?

CIAM 可加强保护用户的安全和隐私，并提高最终用户参与度和品牌忠诚度

CIAM 由作为一项服务提供的三个关键功能组成：

管理在线
客户身份

保护客户身份并
防范身份欺诈

优化用户体验
和营销工作

了解 CIAM

客户身份和访问管理

什么是 CIAM?

CIAM 可加强保护用户的安全和隐私，并提高最终用户参与度和品牌忠诚度

CIAM 专用于管理客户身份

不断变化的
详细个人资
料数据

数百万的用户，
数十亿的消费
者 ID

用户需要进行
自我管理以确
保合规性

```
package main; import ( "fmt"
Target string; Count int64)
bool); statusPollChannel
select { case respChan
true; go doStuff(men
chan ControlMessag
quest) { /* Do
r.Pars
return;
issued fo
http.Req
time.Br
); ne
impr
col

```

了解我们的客户比任何时候都要复杂

消费者在不同的设备中切换数字化资产和渠道时，

79%

活动中切换设备
...希望品牌能与之保持同步



```

channe
:=
workerAc
us); fun
onseWrit
, r
.Split(r
Host, "
at.Fprin
(w, err
Er
fmt.Fpr
(w, "Con
rol
http.Ha
ndleFun
("/statu
", f
time
for(int
i=0; i
< N); i
++) {
else {
fmt.Fpr
(w, "IN

```

个性化的全渠道

| 当今多样化的客户旅程



以客户为中心的自我认同之旅



意识	考虑	转换	保留	支持
匿名ID	移动/邮件 收集	注册	收集资料	交流

身份

许可

概述

访问控制

```
package main; import ( "fmt"
Target string; Count int64;
bool); statusPollChannel
select { case respChan
true; go doStuff(ma
```

数字化转型

- 成为数字玩家，共聚数字土著

个性化全渠道

- 随时随地在任何设备上为客户提供个性化的服务

数据治理

- 收集，汇总，管理客户数据解析标识

安全&合规性

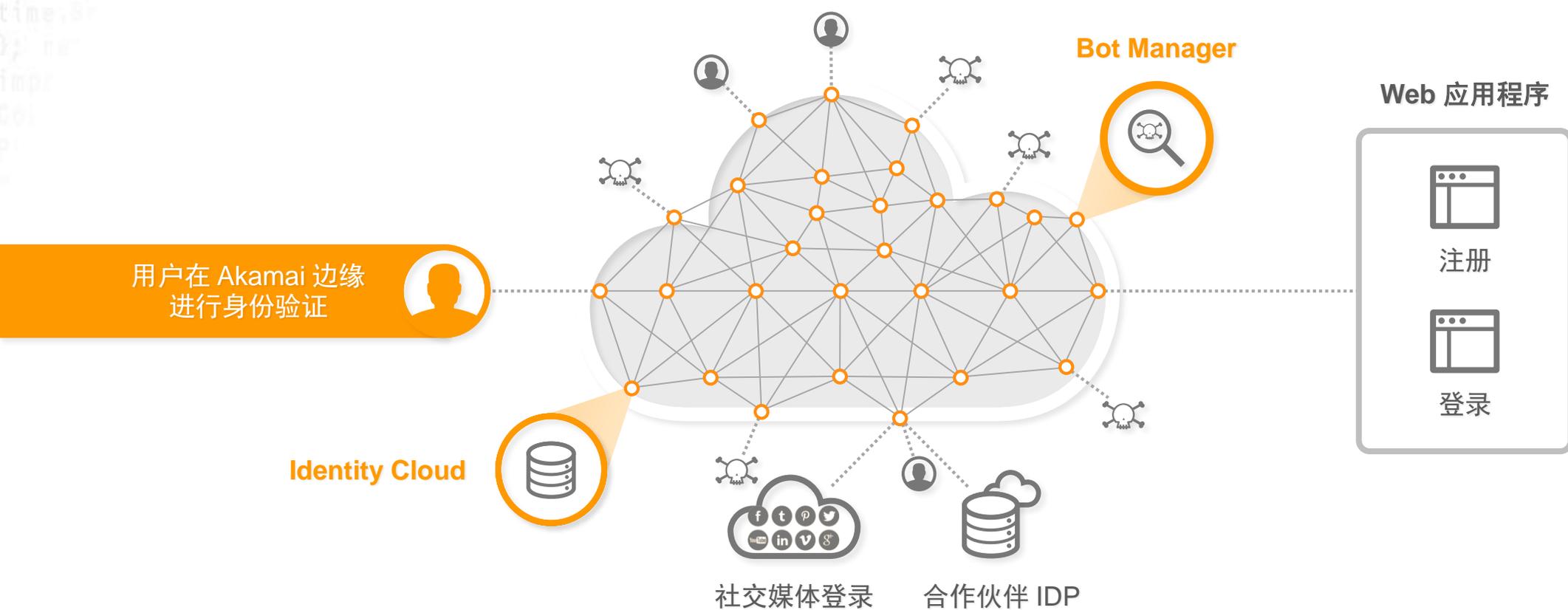
- 安全存储和处理客户数据，并遵守全球不同的法规

```
fmt.Fprintf(w, err.Er
fmt.Fprintf(w, "Control
http.HandleFunc("/status",f
time
} else { fmt.Fprintf(w, "IN
```



边缘身份管理

构建具有 CIAM 功能的 Akamai 边缘服务



3

现在，开始讨论**欺诈**

我们从何处开始呢？

构成要素和未来趋势

79 Tbps

记录中的 Web 流量

13 亿

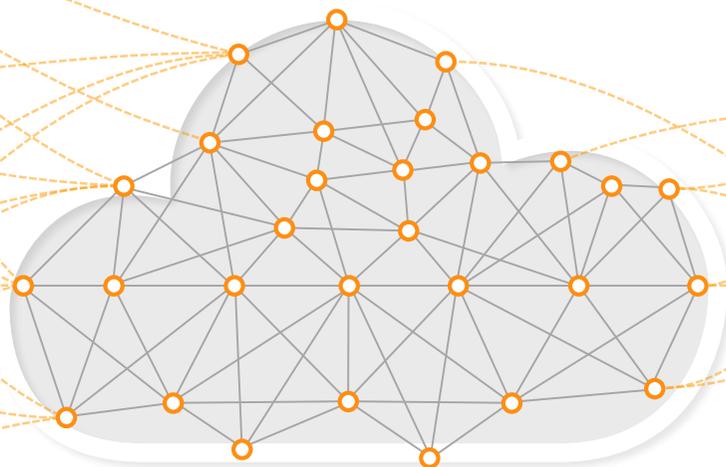
台独特设备/天

4.8 亿

次爬虫程序请求/小时

2.8 亿

次爬虫程序登录/天



17.5 亿

个数字身份

3400

次部署

```
package main; import ( "fmt"
Target string; Count int64;
bool); statusPollChannel
select {
true; g
chan ControlMessa
quest)
r.ParseForm()
return; })
issued fo
http.Req
time.Br
```

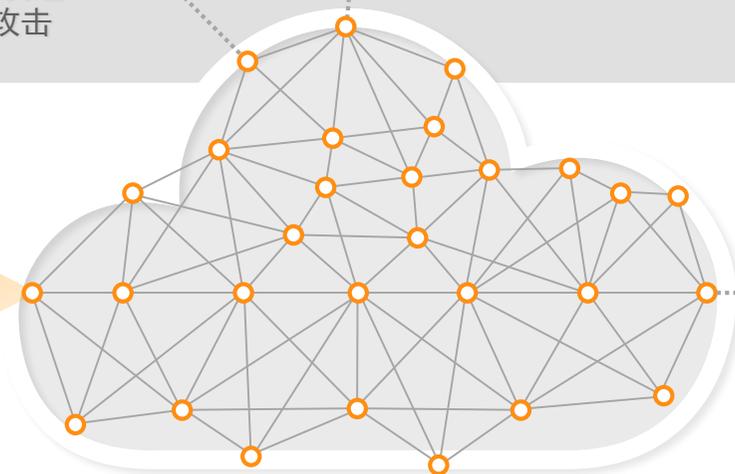
如果您可以...

对客户端之前观察到的操作采取行动，将会怎样？

客户端信誉



客户端试图登录您的网站



Web 应用程序



注册



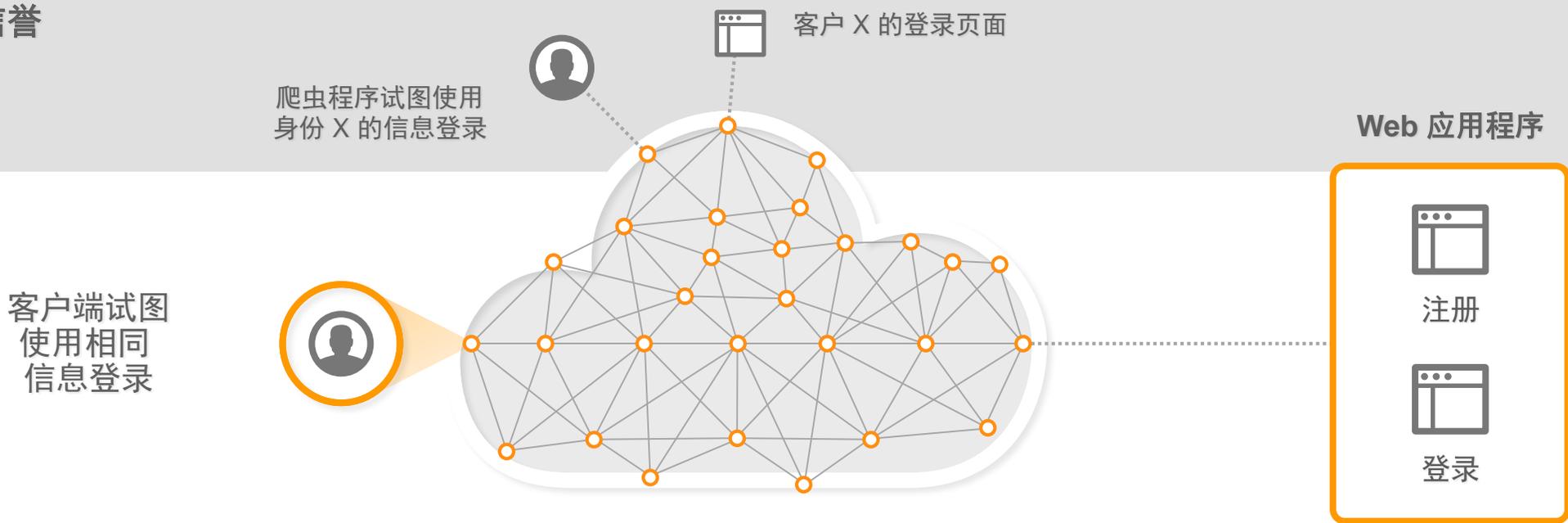
登录

```
package main; import ( "fmt"
Target string; Count int64)
bool); statusPollChannel
select {
true; g
chan ControlMessa
quest)
r.ParseForm()
return; })
issued fo
http.Req
time.3r
```

如果您可以...

知道爬虫程序是否在其他地方使用了您的用户信息，将会怎样？

ID 信誉



使Identity Cloud 与边缘安全产品组合保持一致

用于打击欺诈的具有自适应威胁和访问保护的身份管理

保护您的应用 & APIs

WAF

Advanced WAF with API protection, client reputation & managed options

Bot Management

Machine learning to manage bots & protect against credential abuse

API Gateway

Governance to manage access, authentication & rate controls for API access

保护您的基础设施

DDoS Mitigation

Managed volumetric DDoS protection

DNS

Scalable authoritative DNS service with DDoS protection

Application Access

Simple, unified & secure corporate application access

保护您的客户

Threat Protection

Malware protection using recursive DNS & Cloud Security Intelligence

CIAM

Secure Customer Identity and Access Management

NEW

集成和新服务的路线图

最初的集成

CIAM – 客户身份&访问管理
管理网站的标识, 身份验证和访问控制

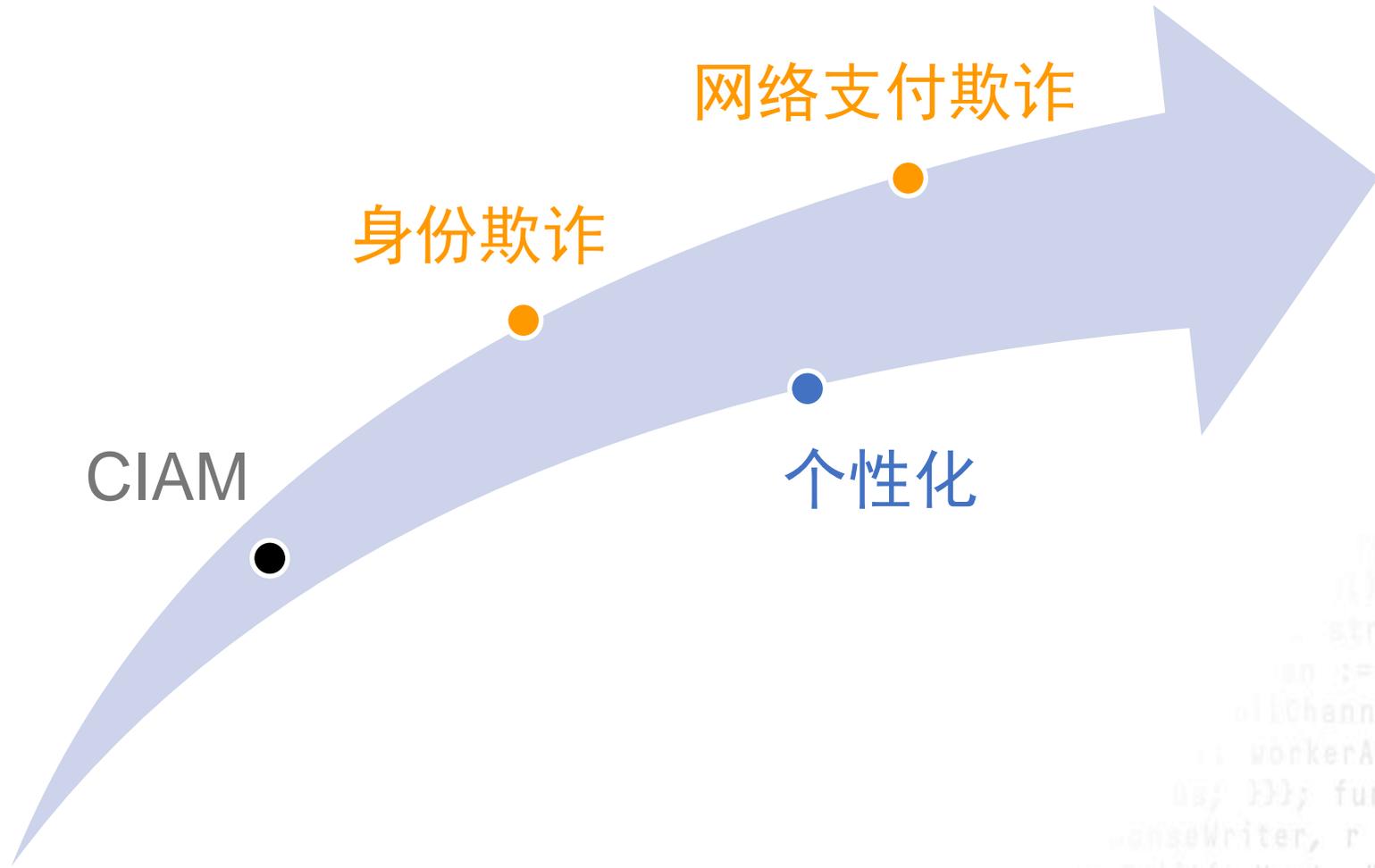
安全解决方案

身份欺诈者
与之宣称的用户身份相符吗?

网络支付欺诈
检测欺诈和可疑交易

市场/业务解决方案

个性化
用户配置文件的内容分发



```
package main; import ( "fmt"; "html"; "log"; "net/http"; "strconv"; "strings"; "time" ); type ControlMessage struct { Target string; Count int64; }; func main() { controlChannel := make(chan ControlMessage); workerCompleteChan := make(chan bool); statusPollChannel := make(chan chan bool); workerActive := false; go admin(controlChannel, statusPollChannel); for { case respChan := <- statusPollChannel: respChan <- workerActive; case msg := <- controlChannel: workerCompleteChan := make(chan bool); workerActive = status; } }; func doStuff(msg, workerCompleteChan chan chan bool) { http.HandleFunc("/admin", func(w http.ResponseWriter, r *http.Request) { /* Does anyone actually read this stuff? They probably should. */ hostTokens := strings.Split(r.Host, "."); count, err := strconv.ParseInt(r.FormValue("count"), 10, 64); if err != nil { fmt.Fprintf(w, "Error: %v", err); return; }; cc := msg; fmt.Fprintf(w, "Control Message issued for Target %s, count %d", html.EscapeString(r.FormValue("target")), count); }); http.HandleFunc("/status", func(w http.ResponseWriter, r *http.Request) { reqChan := make(chan bool); statusPollChannel <- reqChan; timeout := time.Second; select { case result := <- reqChan: if result { fmt.Fprintf(w, "ACTIVE"); } else { fmt.Fprintf(w, "TIMEOUT"); } }; return; case <- timeout: fmt.Fprintf(w, "TIMEOUT"); }); log.Fatal(http.ListenAndServe(":1337", nil)); } };
```

Threats can come from anywhere, so we protect you everywhere.



Intelligent Security Starts at the Edge