

Repsheet

A Behavior Based Approach to Web Application Security



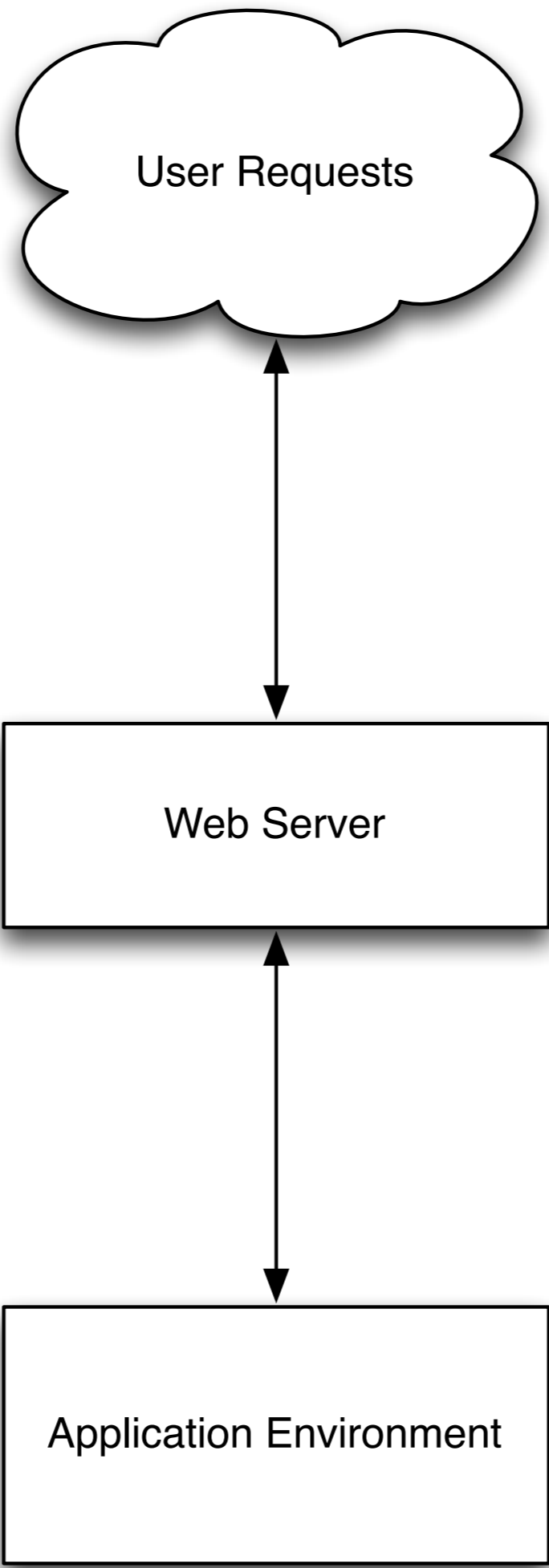
Aaron Bedra
Application Security Lead
Braintree Payments

**Right now, your web
applications are being
attacked**

And it will happen
again, and again, and
again

**But not always in the
way you think**

Let's take a look at
typical application
security measures



Username

Password

Remember Me

[Forgot password?](#)

LOGIN



roland : 12345



roland : 12345



And we go on with our
day

**How many of you stop
there?**

**It's time to start asking
more questions**

But remember...

**Don't impact user
experience!**

???

- Signature based detection
- Anomaly detection
- Reputation based intelligence
- Action
- Repsheet

Signatures

ModSecurity

Web Application Firewall

Rule based detection

**Allows you to block or
alert if traffic matches a
signature**

Improved by the OWASP Core Rule Set

**A great tool to add to
your stack**

**Works with Apache,
nginx, and IIS**

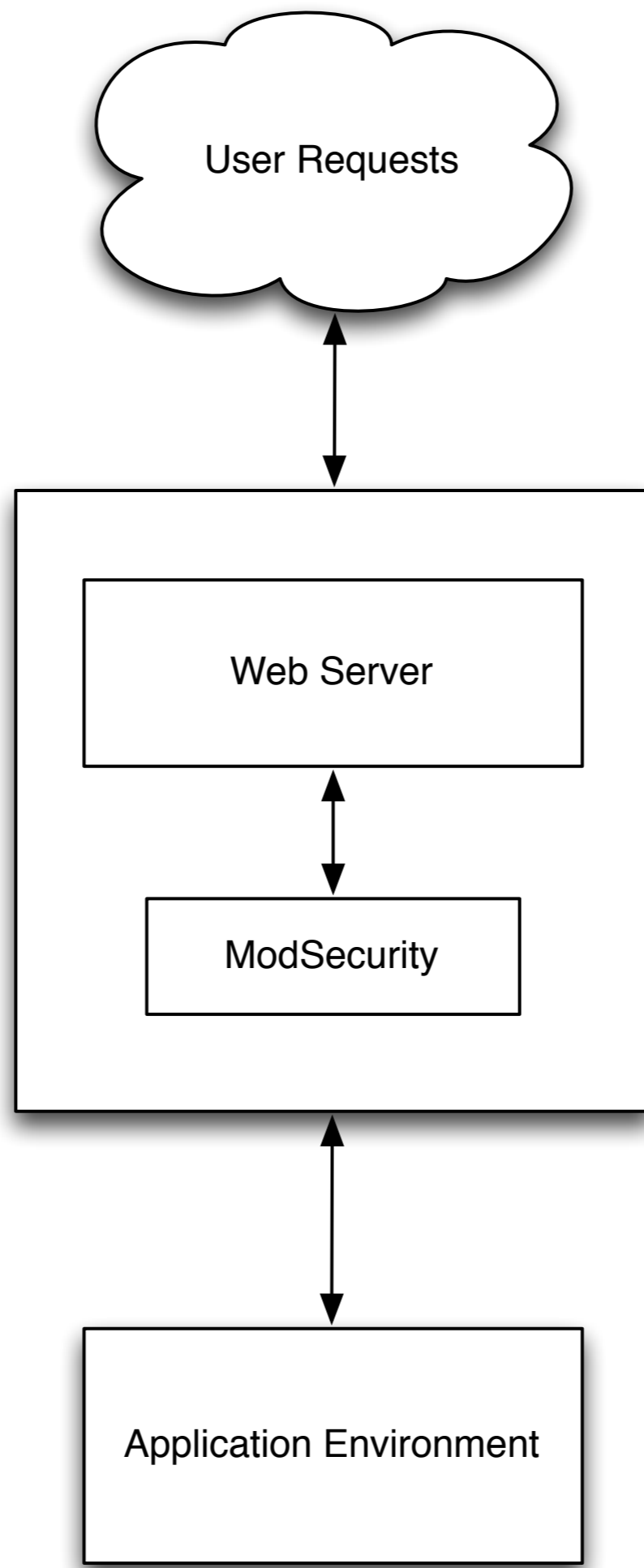
Works well with Apache

**Like most signature
based tools it requires
tuning**

And has a high
possibility of false
positives

**Great for helping with
0-day attacks**

**Favor alerting over
blocking in most
scenarios**



Anomalies

```
10.20.253.8 - - [23/Apr/2013:14:20:21 +0000]  
"POST /login HTTP/1.1" 200 267 "-" "Mozilla/  
5.0 (Windows NT 6.1; WOW64; rv:8.0) Gecko/  
20100101 Firefox/8.0" "77.77.165.233"
```



```
10.20.253.8 - - [23/Apr/2013:14:20:22 +0000]  
"POST /users/king-roland/credit_cards HTTP/  
1.1" 302 2085 "-" "Mozilla/5.0 (Windows NT  
6.1; WOW64; rv:8.0) Gecko/20100101 Firefox/  
8.0" "77.77.165.233"
```

```
10.20.253.8 - - [23/Apr/2013:14:20:23 +0000]  
"POST /users/king-roland/credit_cards HTTP/  
1.1" 302 2083 "-" "Mozilla/5.0 (Windows NT  
6.1; WOW64; rv:8.0) Gecko/20100101 Firefox/  
8.0" "77.77.165.233"
```

```
10.20.253.8 - - [23/Apr/2013:14:20:24 +0000]
"POST /users/king-roland/credit_cards HTTP/
1.1" 302 2085 "-" "Mozilla/5.0 (Windows NT
6.1; WOW64; rv:8.0) Gecko/20100101 Firefox/
8.0" "77.77.165.233"
```

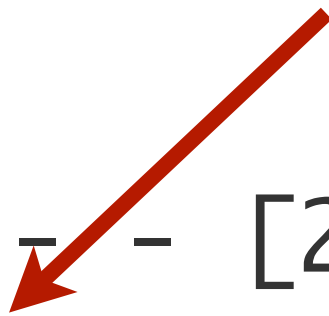
What do you see?

I see a website getting
carded

???

Play by play


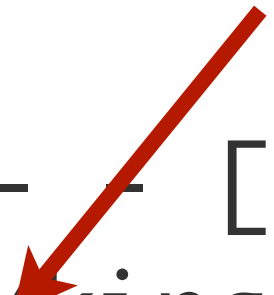
Login Request



```
10.20.253.8 - - [23/Apr/2013:14:20:21 +0000]  
"POST /login HTTP/1.1" 200 267 "-" "Mozilla/  
5.0 (Windows NT 6.1; WOW64; rv:8.0) Gecko/  
20100101 Firefox/8.0" "77.77.165.233"
```


Add credit card to account #1


1 sec delay



```
10.20.253.8 - - [23/Apr/2013:14:20:22 +0000]
"POST /users/king-roland/credit_cards HTTP/
1.1" 302 2085 "-" "Mozilla/5.0 (Windows NT
6.1; WOW64; rv:8.0) Gecko/20100101 Firefox/
8.0" "77.77.165.233"
```

Add credit card to account #2

1 sec delay



```
10.20.253.8 - - [23/Apr/2013:14:20:23 +0000]
"POST /users/king-roland/credit_cards HTTP/
1.1" 302 2083 "-" "Mozilla/5.0 (Windows NT
6.1; WOW64; rv:8.0) Gecko/20100101 Firefox/
8.0" "77.77.165.233"
```



FF 8 on Windows 7
or Bot?

Add credit card to account #3

```
10.20.253.8 - - [23/Apr/2013:14:20:24 +0000]  
"POST /users/king-roland/credit_cards HTTP/  
1.1" 302 2085 "-" "Mozilla/5.0 (Windows NT  
6.1; WOW64; rv:8.0) Gecko/20100101 Firefox/  
8.0" "77.77.165.233"
```

Plovdiv Bulgaria

1 sec delay

FF 8 on Windows 7
or Bot?

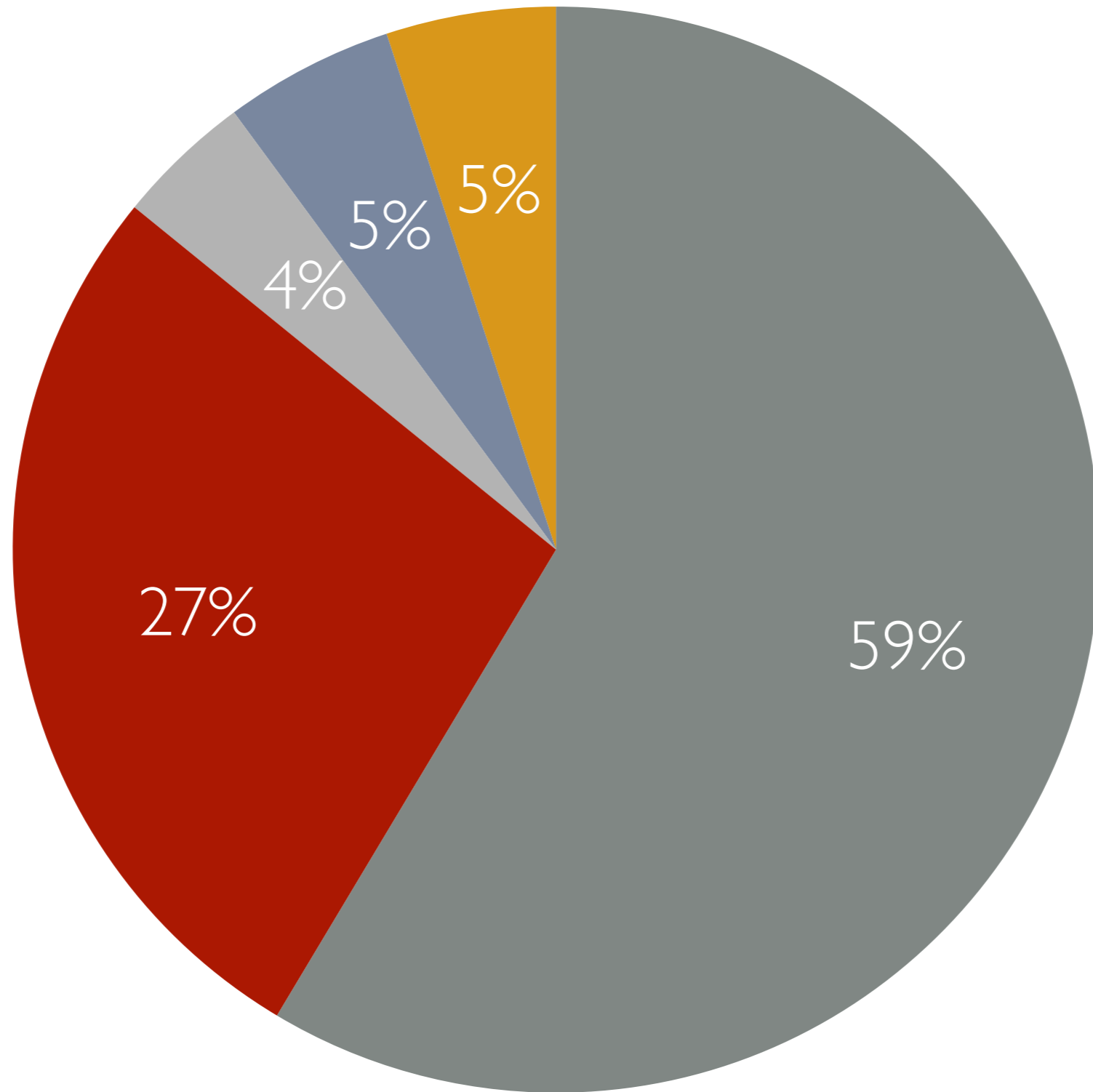
And this continues...

10,000 more times

**Those were the only
requests that IP address
made**

Aside from the number
of requests what else
gave it away?

● GET ● POST ● HEAD ● PUT ● DELETE



**HTTP method
distribution is
important**

**When an actor deviates
significantly, there must
be a reason!**

Let's talk GeolP

Adding GeolP
information is
generically useful

**But it also helps in the
face of an attack**

**It can help protect you
and your users**

Scenario

**King Roland gets his
GMail account hacked**

**Hacker sends a
password reset request
to your server**

**Normally, you would
email the reset**

Unless...

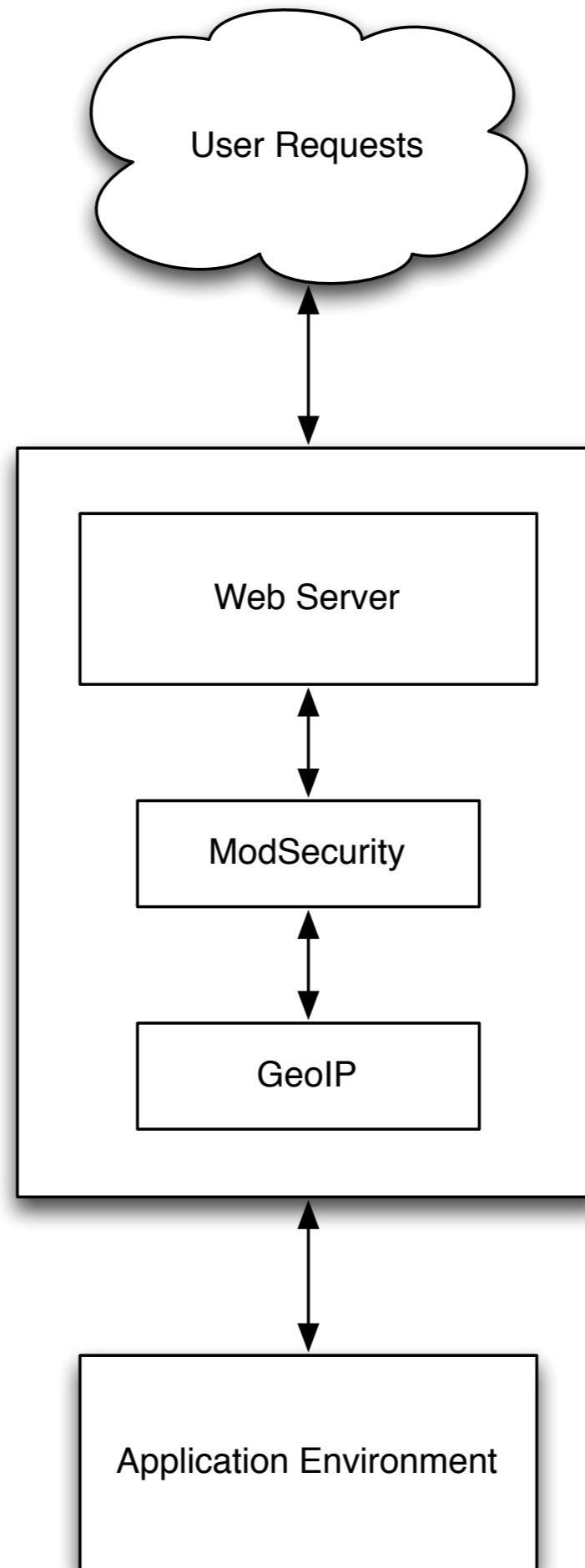
You realize that King
Roland always logs in
from Druidia

But the hacker is
requesting the reset
from Spaceball City

**Instead of sending the
reset, you now ask
some questions**

**And hopefully protect
King Roland from
further bad actions**

**GeoIP detection also
helps you block traffic
from unwanted
countries**



Other Anomalies

- Request rate
- Header ordering
- TCP Fingerprint vs. User Agent
- Account Create/Delete/Subscribe
- Anything you can imagine

**What do they have in
common?**

**Does the behavior fit
an equation?**

**If so, your detection is
simple**

**Request rate >
Threshold**

**TCP fingerprint !=
User Agent**

**But the HTTP method
deviation is harder**

**100% GET requests
with a known UA (e.g.
Google) is ok**

**100% POST requests is
not**

**But it's not always that
simple**

Scenario

**A high rate of account
create requests are
coming from a single
address**

**Is it a NATed IP or a
fraud/spam bot?**

**We have patterns and
data...**

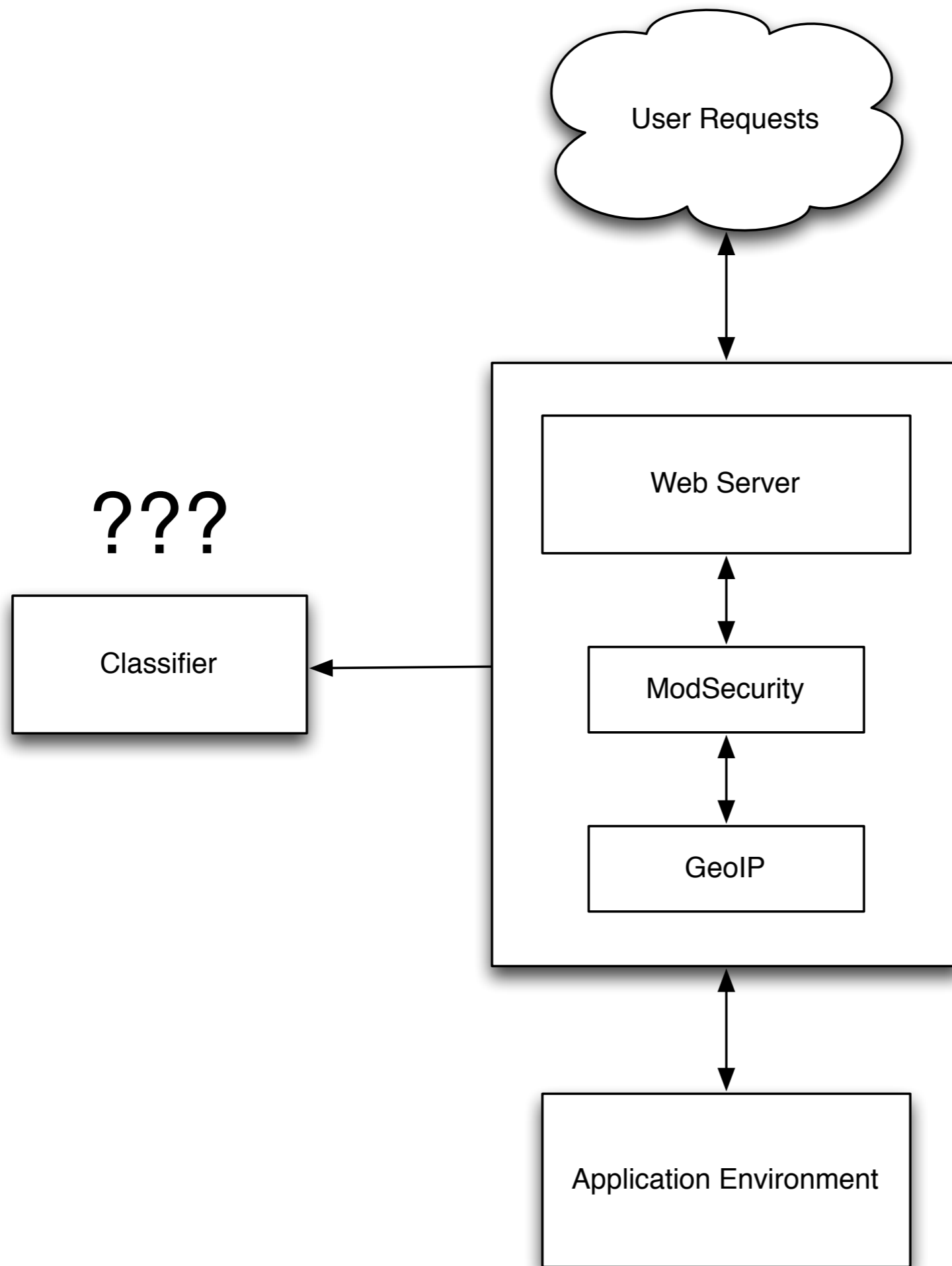
What's the next step?

Quantitative Analysis

~~Quantitative Analysis~~

Security as a Data Science Problem

We can apply some
machine learning to the
data in an attempt to
classify it



**This is where a lot of
the value comes from**

**And combined with
signature detection
helps correlate attack
events**

**But you still need a way
to keep track of it all**

Reputation Based Intelligence

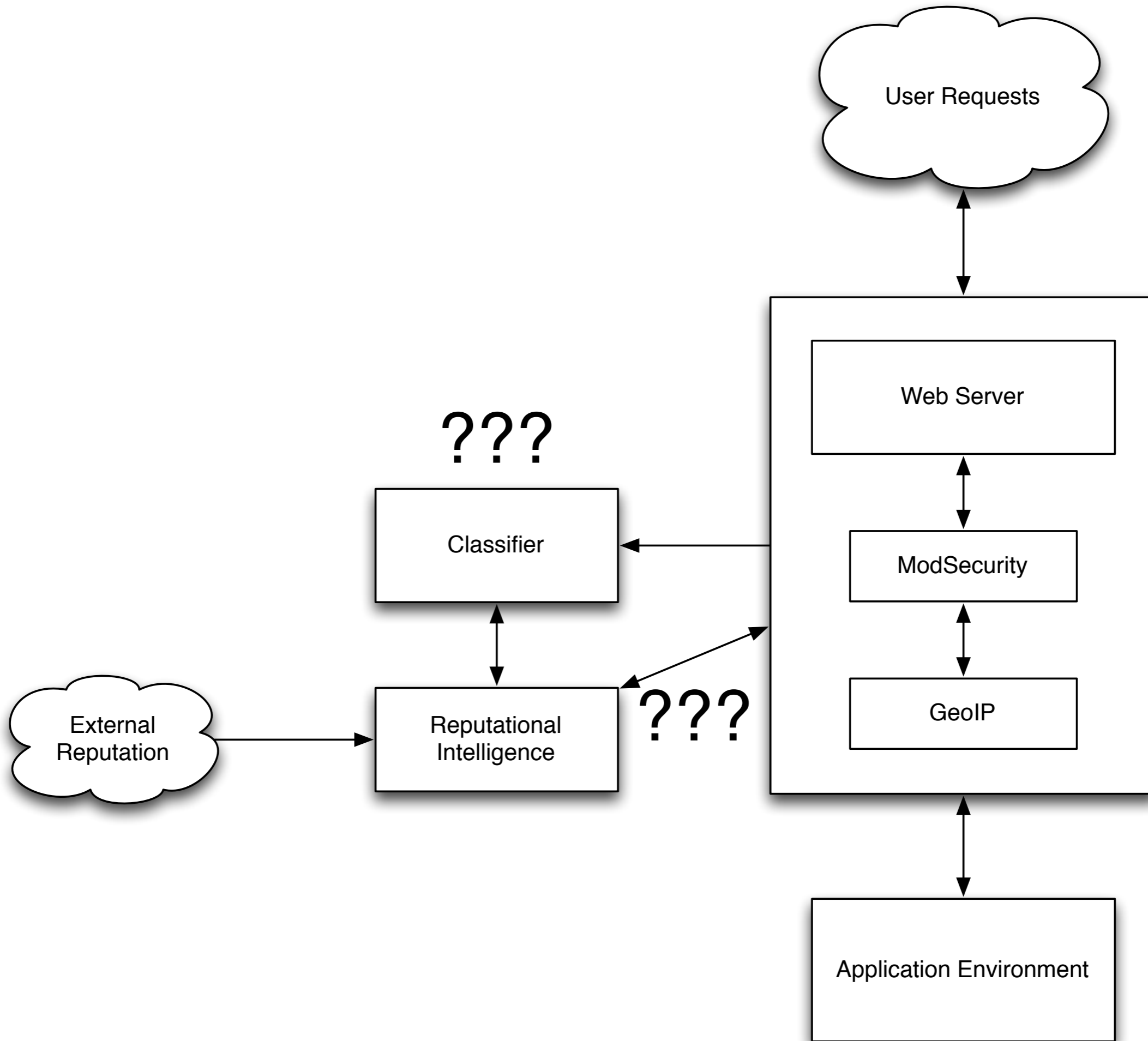
Who's naughty and
who's really naughty

**Built up from the tools/
techniques mentioned
previously**

**Provides local
reputation**

**You can also purchase
external reputation
feeds**

**The combination gives
you solid awareness of
bad actors**



Action

**So now you have a ton
of new information**

What do you do with
it?

Options

- Block the traffic
- Honeypot the attacker
- Modify your response
- Attack back
- Contact the authorities

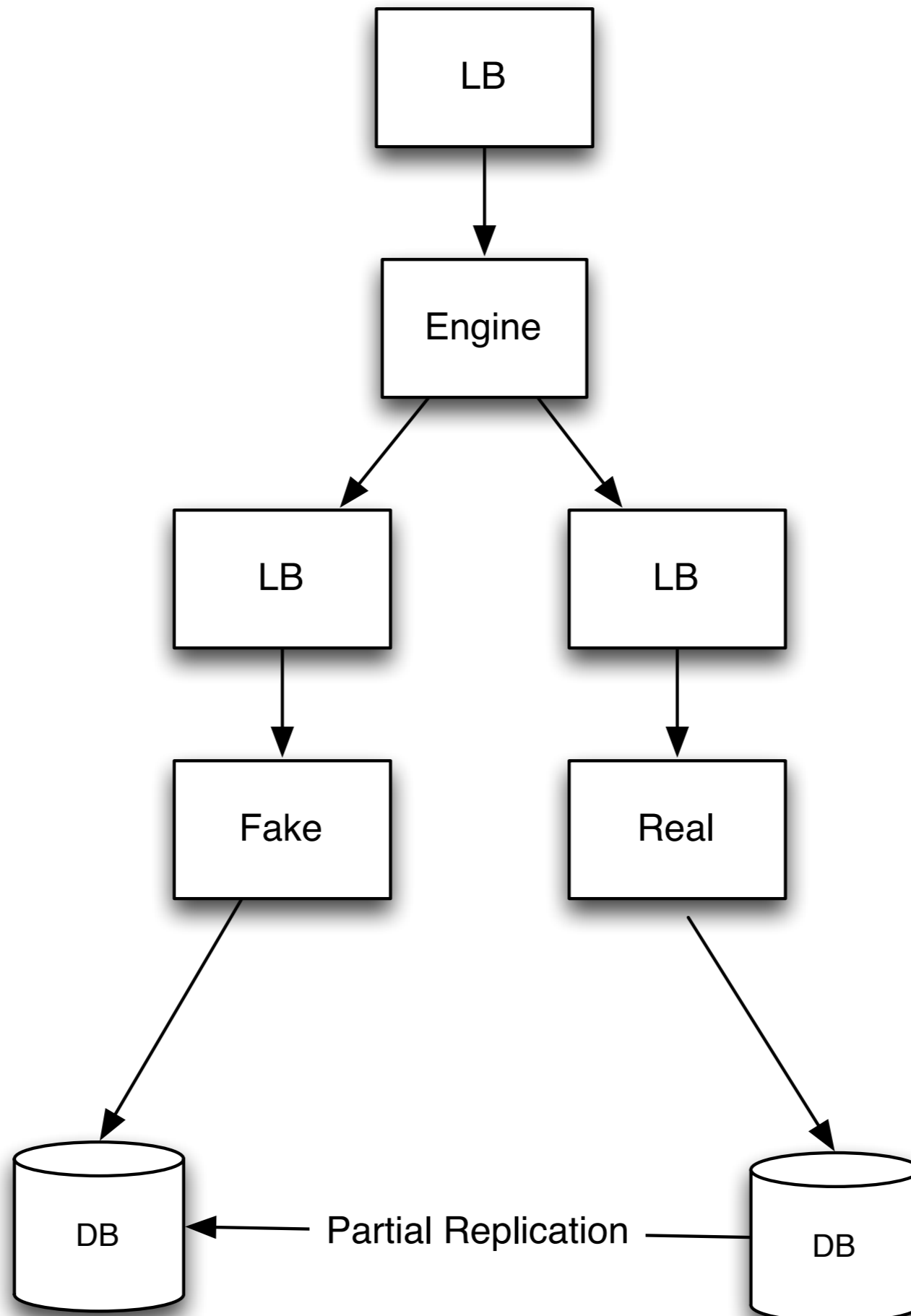
**Blocking the traffic is
straight forward**

**Block at the web server
level (403)**

Block at the firewall level

**Both have advantages/
disadvantages**

**Honeypots are much
more interesting**

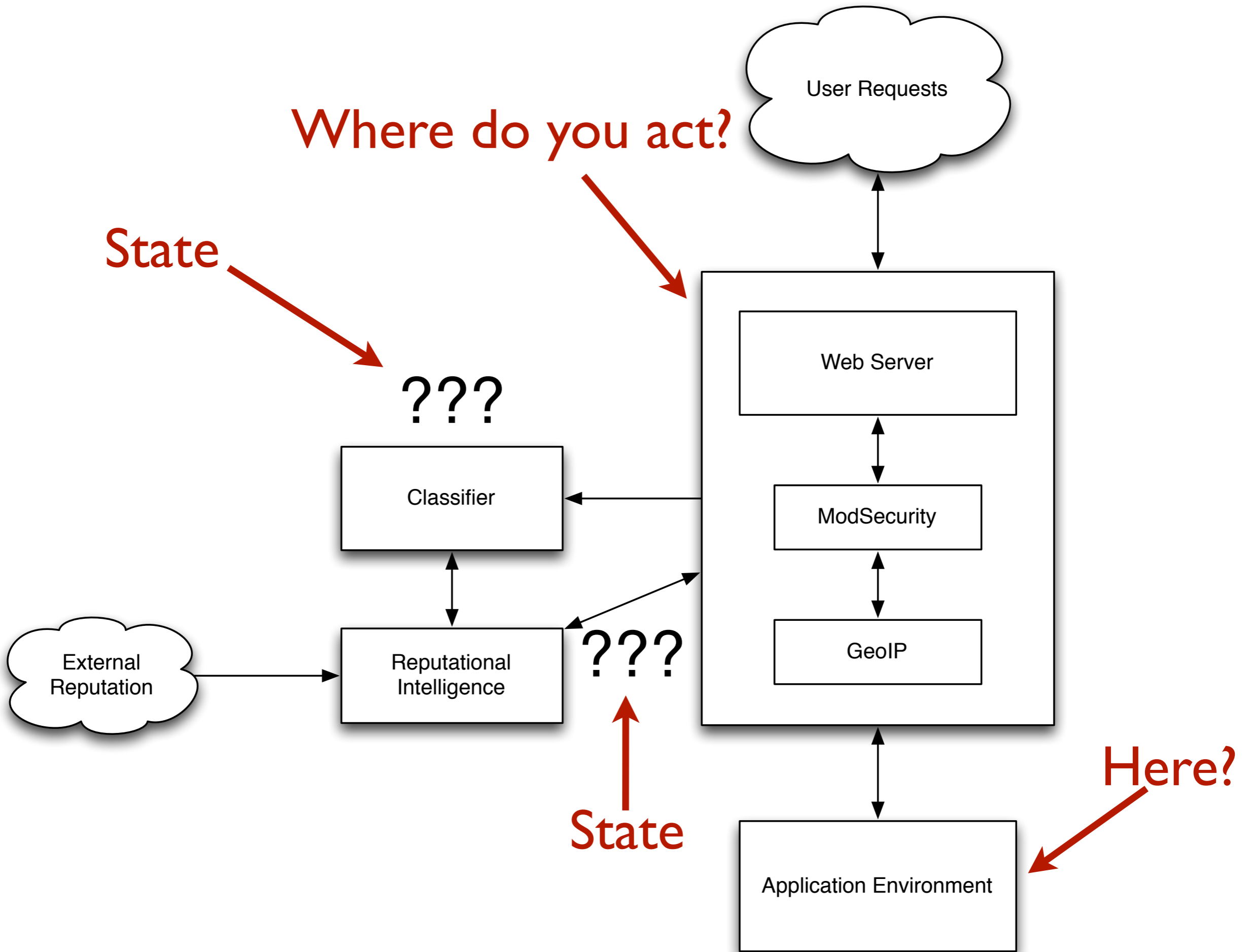


When you honeypot,
the attacker doesn't
know they've been
caught

And it allows you to
study their behavior

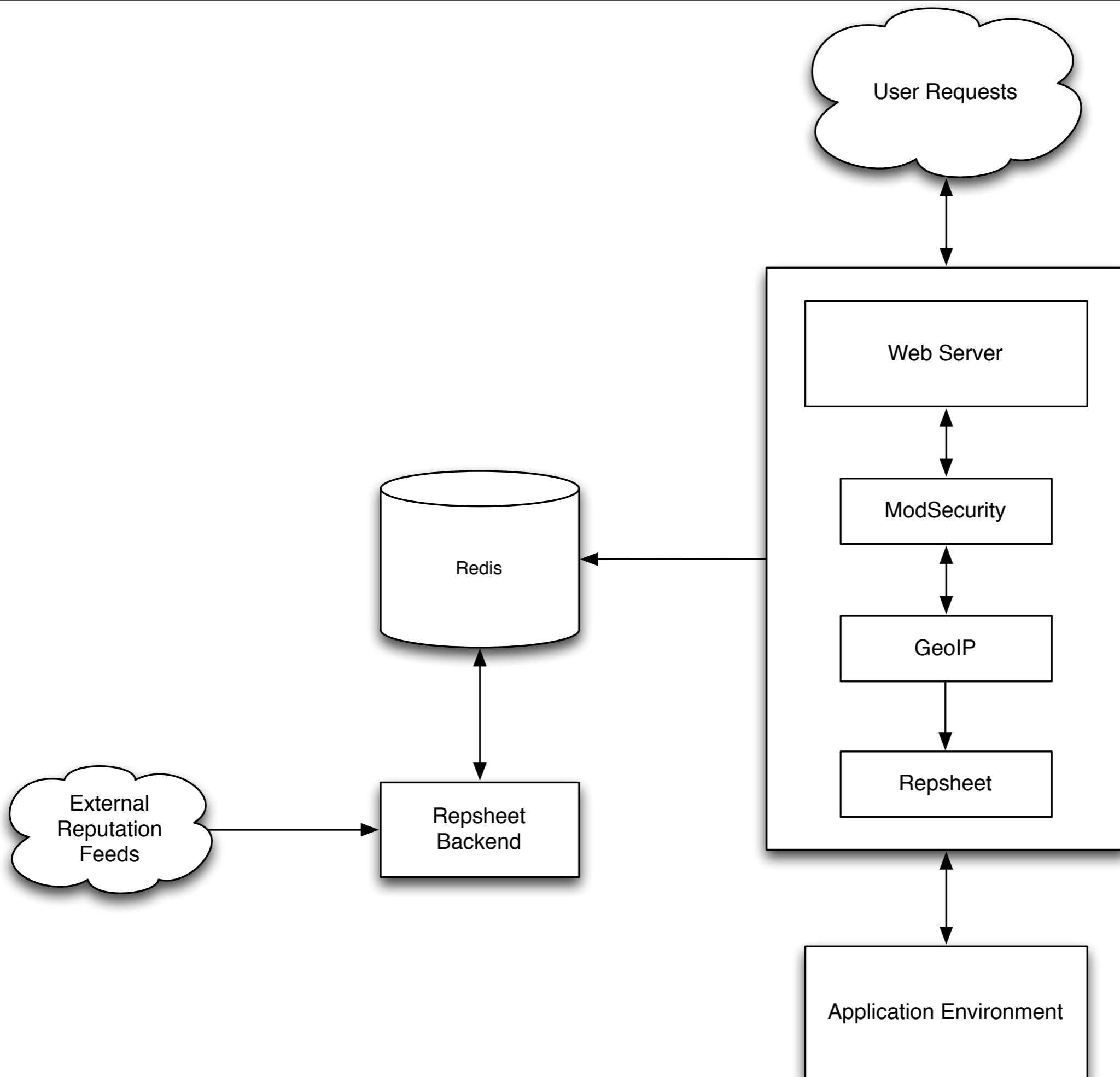
**And update your
approach to preventing
attacks**

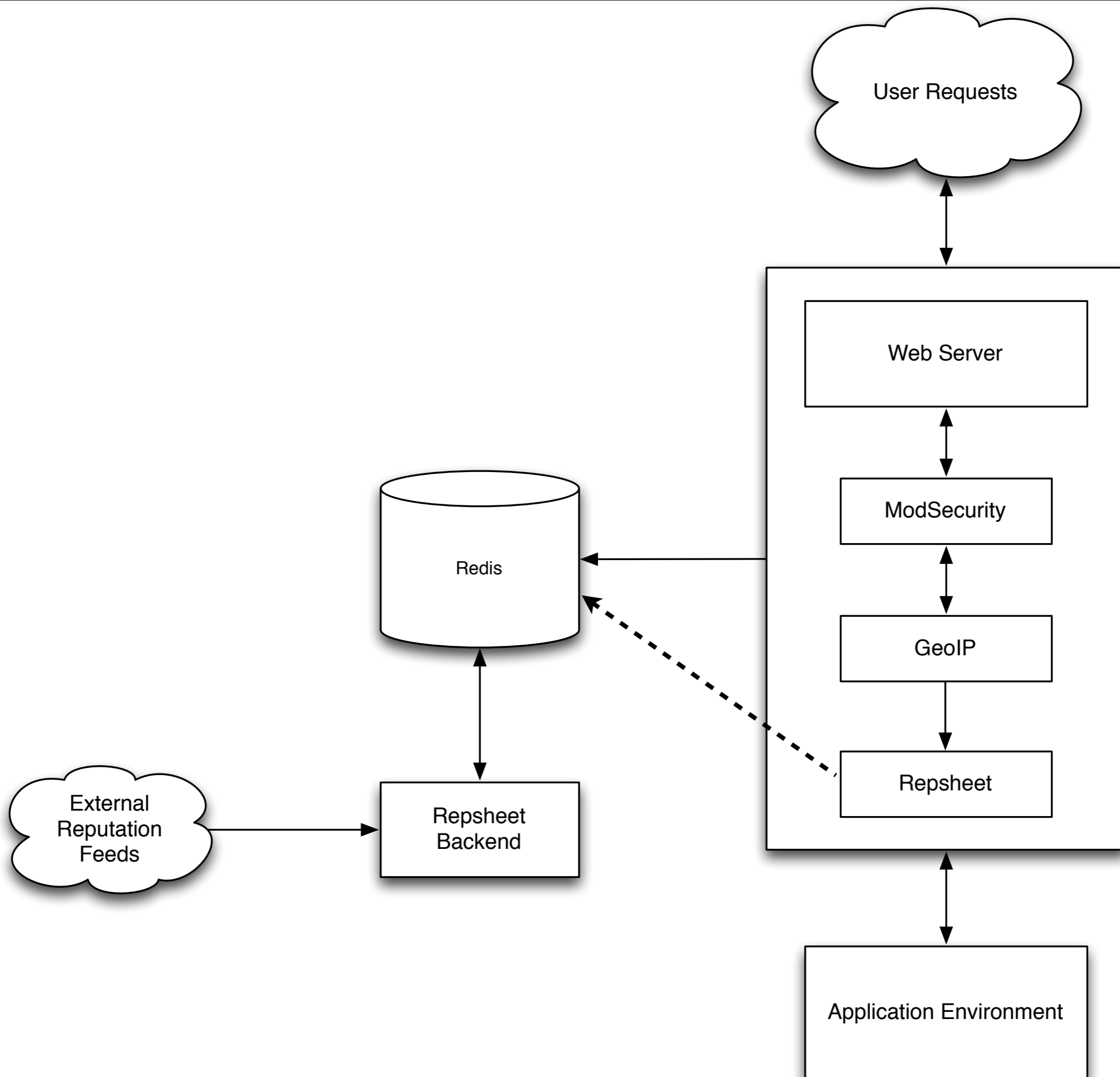
**But all of this requires a
way to manage state
and act on bad behavior**

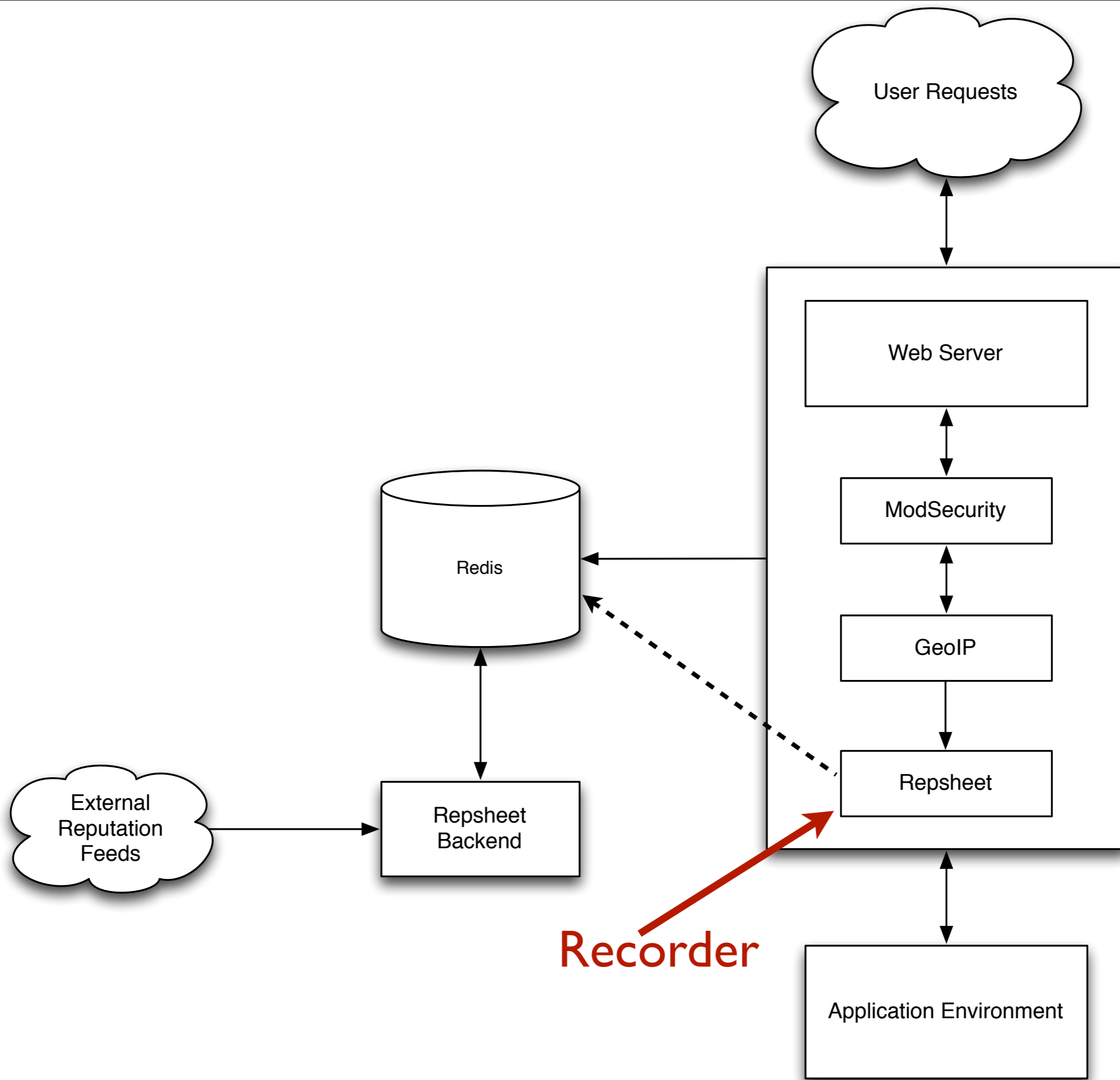


Repsheet

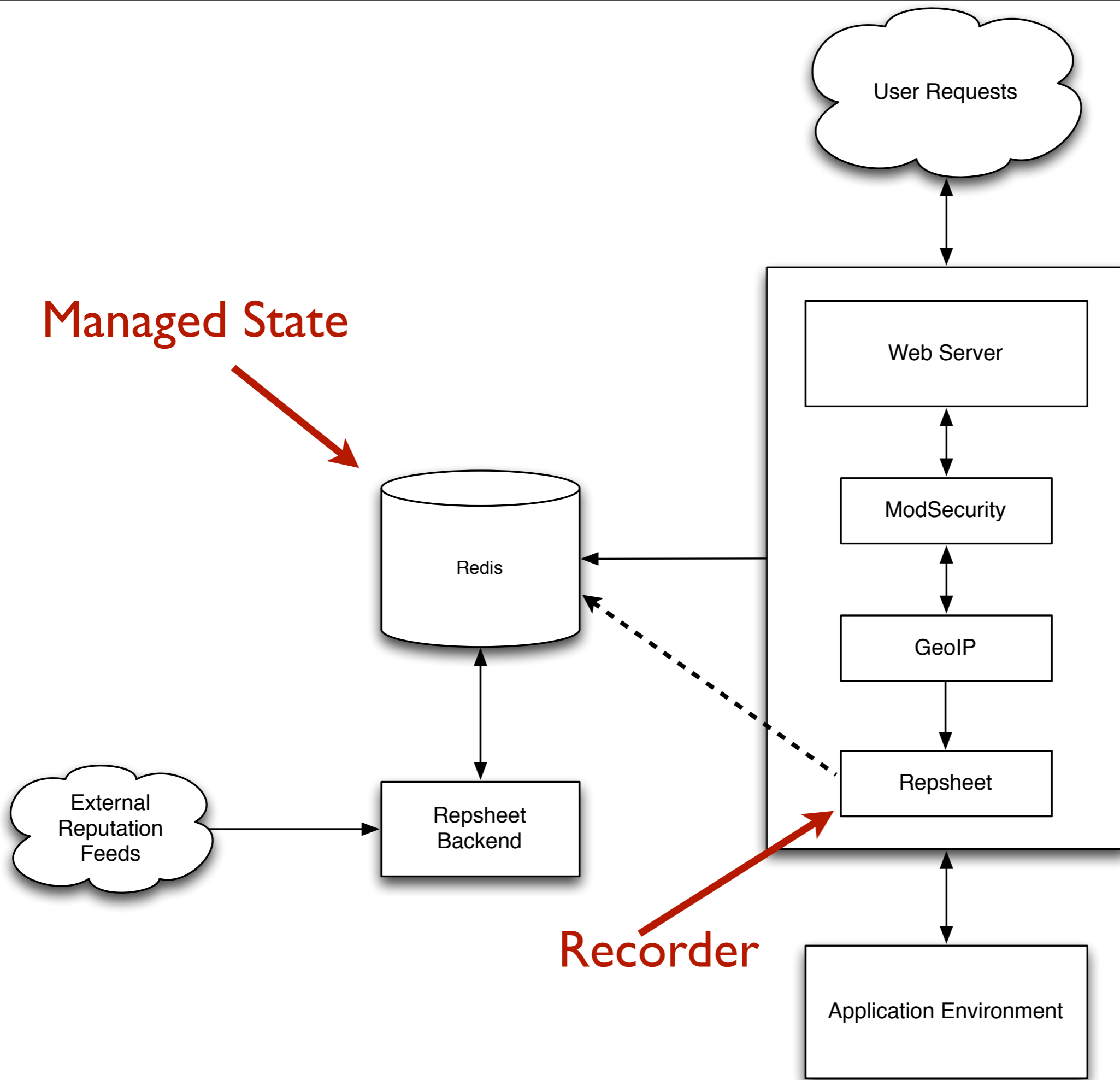
Reputation Engine

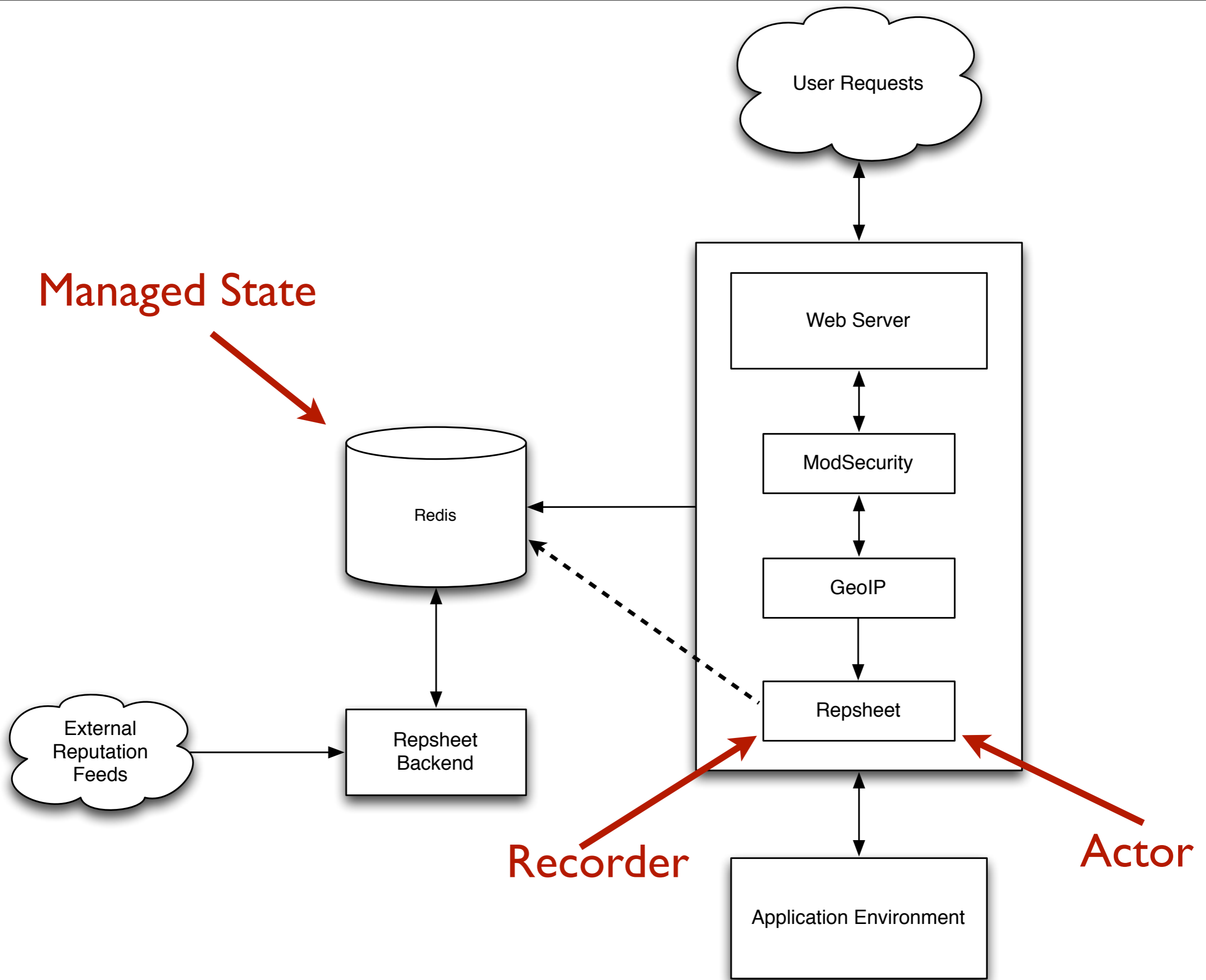


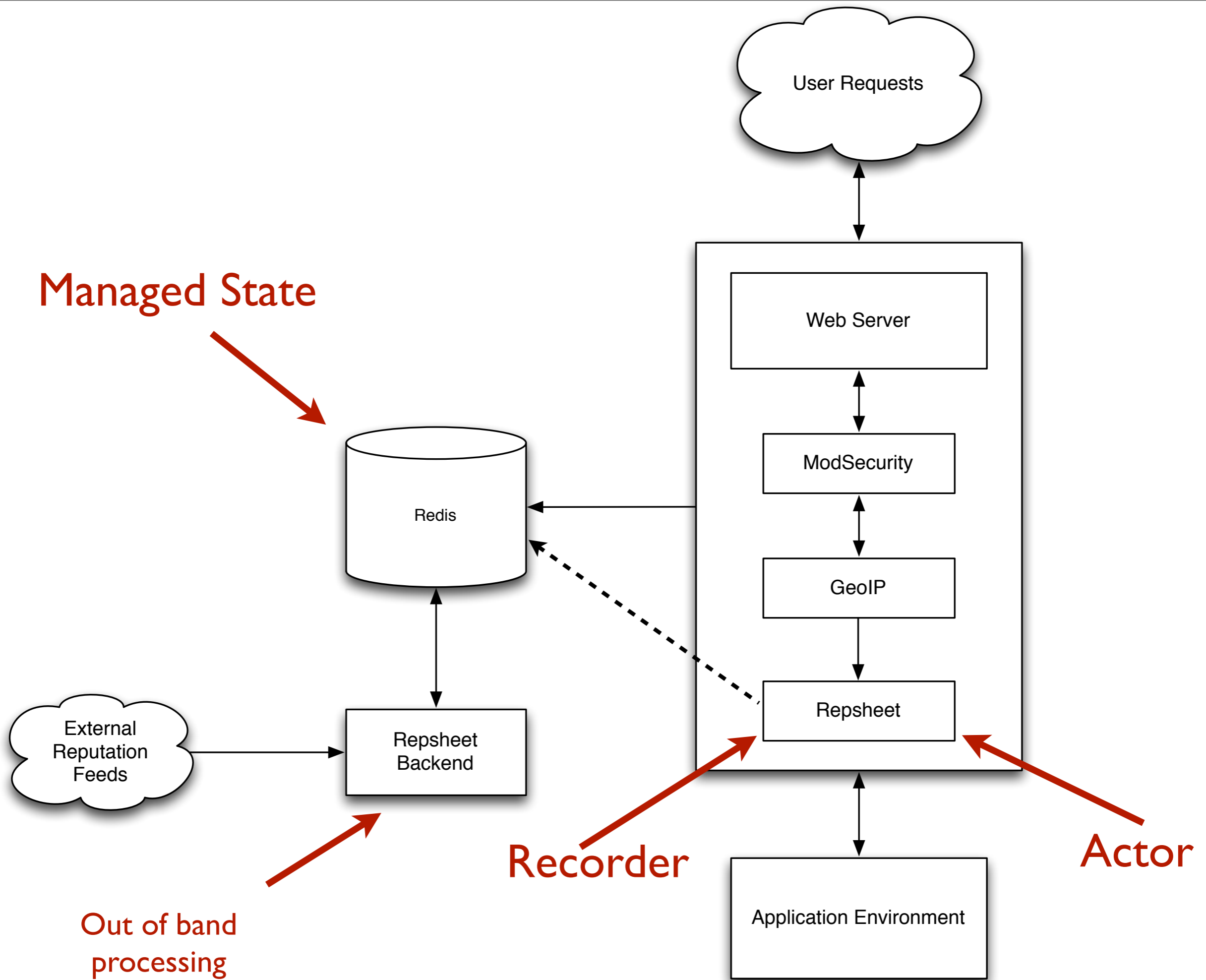




Recorder







Find Activity For an IP

Top 10 Suspect Actors

IP	Triggered Rules	Offenses	Activity	Action
5.9.244.130	981246, 981318, 973336, 973337, 960020, 960024, 950109, 950901	1834	Click to see activity	blacklist
204.232.241.139	973315, 958051, 973307, 958052, 981318, 973336, 960020, 500000, 960024, 973300, 960035, 950901, 981242	625	Click to see activity	blacklist
198.239.178.118	981172, 981231, 981318	168	Click to see activity	blacklist
71.198.4.77	981319	156	Click to see activity	blacklist
50.16.52.137	990012	152	Click to see activity	blacklist
64.236.139.254	981319	95	Click to see activity	blacklist
54.226.166.36	990012	69	Click to see activity	blacklist
110.142.71.71	960020	47	Click to see activity	blacklist
182.188.182.135	960024, 500000, 981245, 958404, 981318, 973337, 973338	46	Click to see activity	blacklist
106.78.4.109	960024, 500000, 981245, 981318, 973337, 973338	44	Click to see activity	blacklist

Blacklisted Actors

IP	Triggered Rules	Offenses	Activity	Action
119.155.8.131	981244, 958051, 973307, 958052, 981245, 981246, 707385, 981247, 981249, 500000, 950001, 950002, 960020, 950120, 950103, 950005, 950006, 960024, 950007, 950901, 981231, 950109, 981250, 960901, 981251, 973331, 973332, 973314, 973333, 973315, 981254, 981272, 981317, 981255, 981318, 981319, 959071, 973336, 981257, 959073, 981276, 981277, 950010, 960032, 850720, 973300, 960035, 981320, 981240, 960911, 981242	23567	Click to see activity	allow
84.235.91.39	973315, 958051, 981245, 981318, 973336, 973338, 000632, 000660, 000680, 000663, 000654, 000628, 000692, 000639, 000685, 990002, 960024, 973300, 981240, 950901, 973331, 981260	1122	Click to see activity	allow
112.210.105.125	960032, 990002, 990012, 960035, 960010	786	Click to see activity	allow
46.4.94.143	973315, 958051, 958052, 973307, 981318, 973336, 500000, 960024, 973300, 950901, 981242	505	Click to see activity	allow
218.104.226.18	950005, 960024, 960017, 981231, 960901, 950103	12	Click to see activity	allow
211.212.39.180	981318, 981172, 981245, 981246, 981319,	7	Click to see activity	allow



**Repsheet helps put
everything together**

Web server module
records activity and
looks for offenders in
the cache

It listens to
ModSecurity and adds
offending IPs to its list

**It provides notification
and/or blocking of
offenders**

**Blocking happens at the
web server level**

**But you can send
Repsheet data to your
firewall for TCP level
blocking**

**Notification sends
headers to the
downstream application**

Which allows each app
to chose how it is going
to respond

**For instance, show a
captcha on signup if
Repsheet alerts**

**Back end looks at the
recorded data for bad
behavior**

**And updates the cache
when it finds offenders**

**You can supply your
own learning models
for the data**

[github.com/repsheet/
repsheet](https://github.com/repsheet/repsheet)

Summary

**There are lots of
indicators of attack in
your traffic**

**Build up a system that
can capture the data
and sort good from bad**

Tools

- ModSecurity
- GeoIP
- Custom rules (velocity triggers, fingerprinting, device id, etc)
- Custom behavioral classification
- Repsheet

And Remember...





Questions?