RSAConference2016

San Francisco | February 29 – March 4 | Moscone Center

Connect to Protect

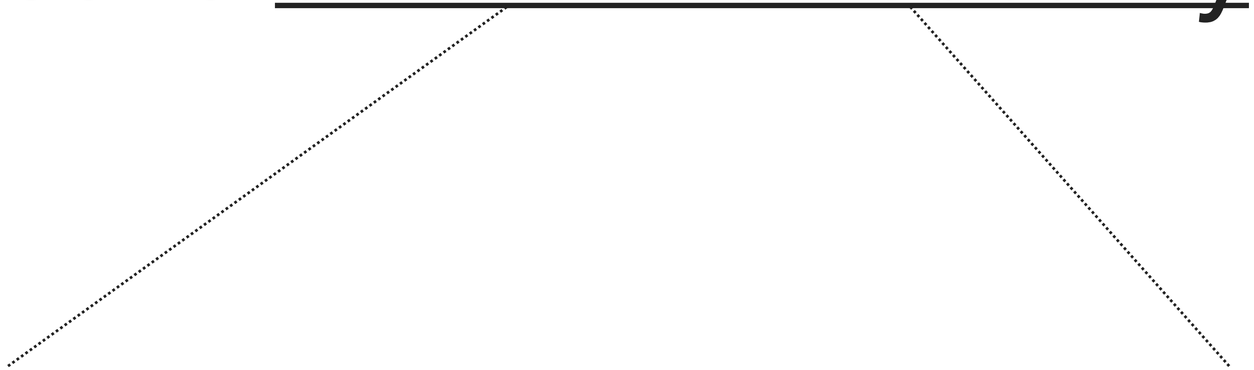
Giving the Bubble Boy an Immune System So He Can Play Outside

Kevin Mahaffey

Founder, CTO
Lookout
@dropalltables

0x00: Why?

Many try to build data-driven security processes.



Some are successful.

Some are unsuccessful.

Data

Cyber

Cyber

Cyber

Machine Learning

Data

Machine Learning

AI

Cyber

AI

Cyber

Data

Data

AI

Data

Machine Learning

Data

Cyber

AI

AI

Machine Learning

Machine Learning

AI

…before man reaches the moon, mail will be delivered within hours from New York to California, to Britain, to India or Australia by guided missiles.

We stand on the threshold of rocket mail.

Arthur E. Summerfield

THE POSTMASTER GENERAL

WASHINGTON

U.S.S. BARBERO
JUN
8
9.30AM
1959

FIRST OFFICIAL
MISSILE
MAIL

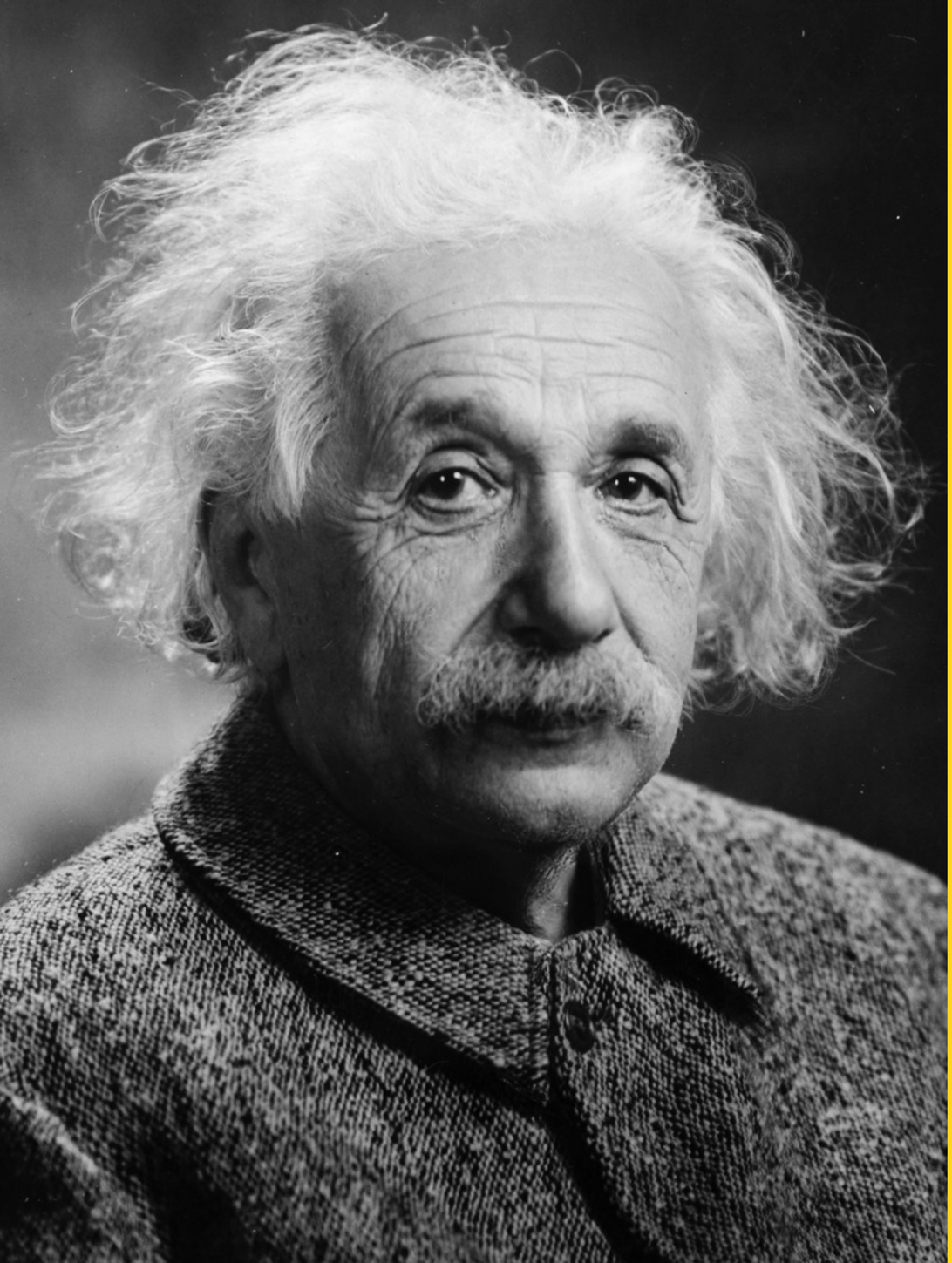

SPACEX

Prediction:

Most security architectures will use data to dynamically determine trust.

Not just for decision support.

0x01: Problems to Solve



If I had only one hour
to save the world,

I would spend fifty-five
minutes defining the
problem,

and only five minutes
finding the solution.

Albert Einstein

attributed!

Foreword

Expect lots of examples about data-driven security architectures drawing on experience of myself and others.

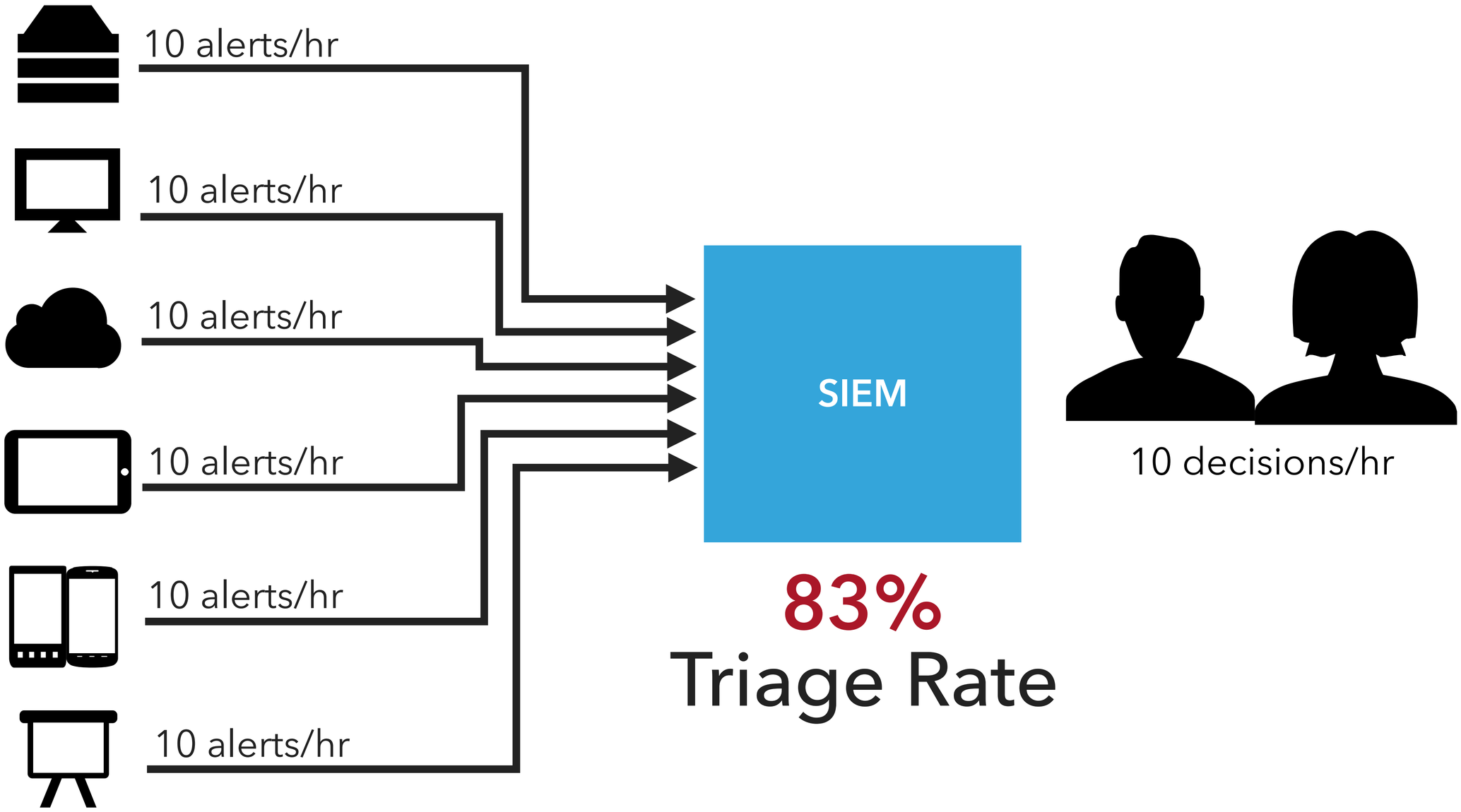
- Large financial institutions
- Large consumer companies
- Product and engineering teams building data-driven security systems

I **will** miss things you think are important

- Let me know what they are: k@lookout.com

Problem 1:

Alerts are scaling faster than humans



83%
Triage Rate


Lightly carbonated.
Serve chilled.
Red Bull
ENERGY DRINK
Red Bull®
ENERGY DRINK
24 x 250 ml
24 x 8.4 FL OZ

Using data more effectively, we should:

Use machines to shoulder some of the load

- …and also react faster than humans

…and have humans just take the hard stuff

…while also distributing decisions outside the security team

…and enable the organization to be more productive.

Problem 2:

Static trust systems don’t adequately reduce risk

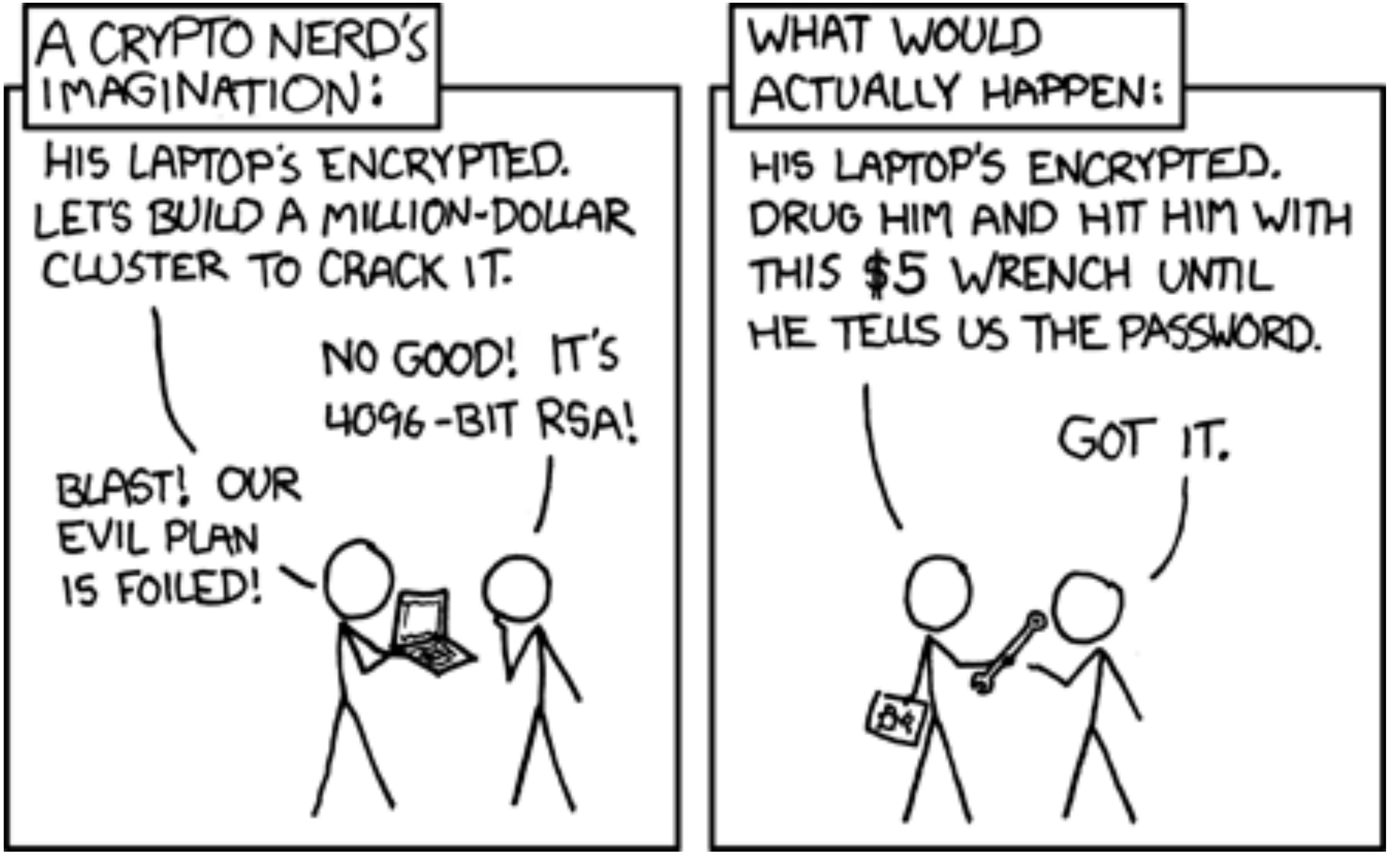
Attackers are misappropriating trust.

VALID auth credentials can be stolen

A VALID internal network connection can be proxied into

A LEGITIMATE host’s integrity can be exploited

A REAL EMPLOYEE can go rogue



http://imgs.xkcd.com/comics/security.png

Attackers evade static trust with knowledge.

Metamorphic malware

- Mutates on every endpoint

“Low and slow” threats that slide under detection thresholds

Environment-aware malware

- e.g. be nice for a while, then do something bad on your target machine

“If you’re sending friend requests that trend to 80% female, thats a red flag, or if you change your birth date a lot–under and above the 18 threshold.”

“When you have single concrete rules, its easy for people to figure them out, but with machine learning, its evolving all the time.”

Joe Sullivan

http://www.forbes.com/forbes/2012/0312/feature-joe-sullivan-site-safety-privacy-facebook-top-cop.html

Theoretical risk mitigation often cannot be put into practice.

Theory: IT controls patch management

- Practice: Mobile, embedded firmware updates are manufacturer-controlled

Theory: Only authorized devices in sensitive network segments

- Practice: 3rd parties (contractors + vendors) behind the firewall

Theory: IT can control network access

- Practice: NAC exceptions + user-workarounds

Theory: IT can inspect all data ingress/egress

- Practice: SSL + pinning + mobile + cloud

Theory: IT can manually whitelist binaries

- Practice: not for BYOD/Vendor/Contractor endpoints (mobile + PC)

Problem 3:

Most architectures aren’t ready

Analytic data != Operational data

Operational: Low-latency, known access patterns, small data

- Real-time transactions

Analytics: High-latency, arbitrary access patterns, large data

- Offline decision support

Too many unintegrated systems

WARNING

SELF-EVOLVING SYSTEM

Source: https://www.flickr.com/photos/arenamontanus

Large Online Payments Company

A few years back: a proposal to build **another** security data system

…but there were already tons of data systems

- Archer
- Metadata platform
- Hadoop
- SIEM

…that evolved organically, and were not integrated

- Some unreliable data adapters (data would show up on one place, but not another)
- No common data models or APIs

https://www.flickr.com/photos/davemorris

Too many granular decisions to manage

WARNING

COGNITIVE HAZARD

Source: https://www.flickr.com/photos/arenamontanus

Granular static policy == lots of decisions == unmanageable

- Firewall rules
- IAM entitlements
- NAC
- Executable whitelists
- AV signatures
- Data classification

[NewCo security widget] increases decision load.

- New products *should* aim to reduce, not increase, decisions

Poor UX

If trust is absolute, you build big walls to keep bad things out

- MFA
- long passwords
- segmented networks

Pain ensues.

Rules

1. The password must be **exactly** 8 characters long.
2. It must contain **at least** one letter, one number, and one of the following special characters.
    a. The **only** special characters allowed are: @ # $
    b. A special chaacter must **not** be located in the first or last position.
3. Two of the same characters sitting next to each other are considered to be a “set.” No “sets” are allowed. **Example: rr, tt**
4. Avoid using names, such as your name, user ID, or the name of your company or employer.
5. Other words that cannot be used are Texas, child, and the months of the year.
6. A new password cannot be too similar to the previous password.
    a. **Example:** previous password - abc#1234; unacceptable new password - acb$1243
    b. Characters in the first, second, and third positions cannot be identical. (abc*****)
    c. Characters in the second, third, and fourth positions cannot be identical. (*bc#****)
    d. Characters in the sixth, seventh, and eighth positions cannot be identical. (*****234)
7. A password can be changed voluntarily (no Help Desk assistance needed) once in a 15-day period. If needed, the Help Desk can reset the password at any time.
8. The previous 8 passwords cannot be reused.

One way to create a password is creative spelling and substitution. Examples:

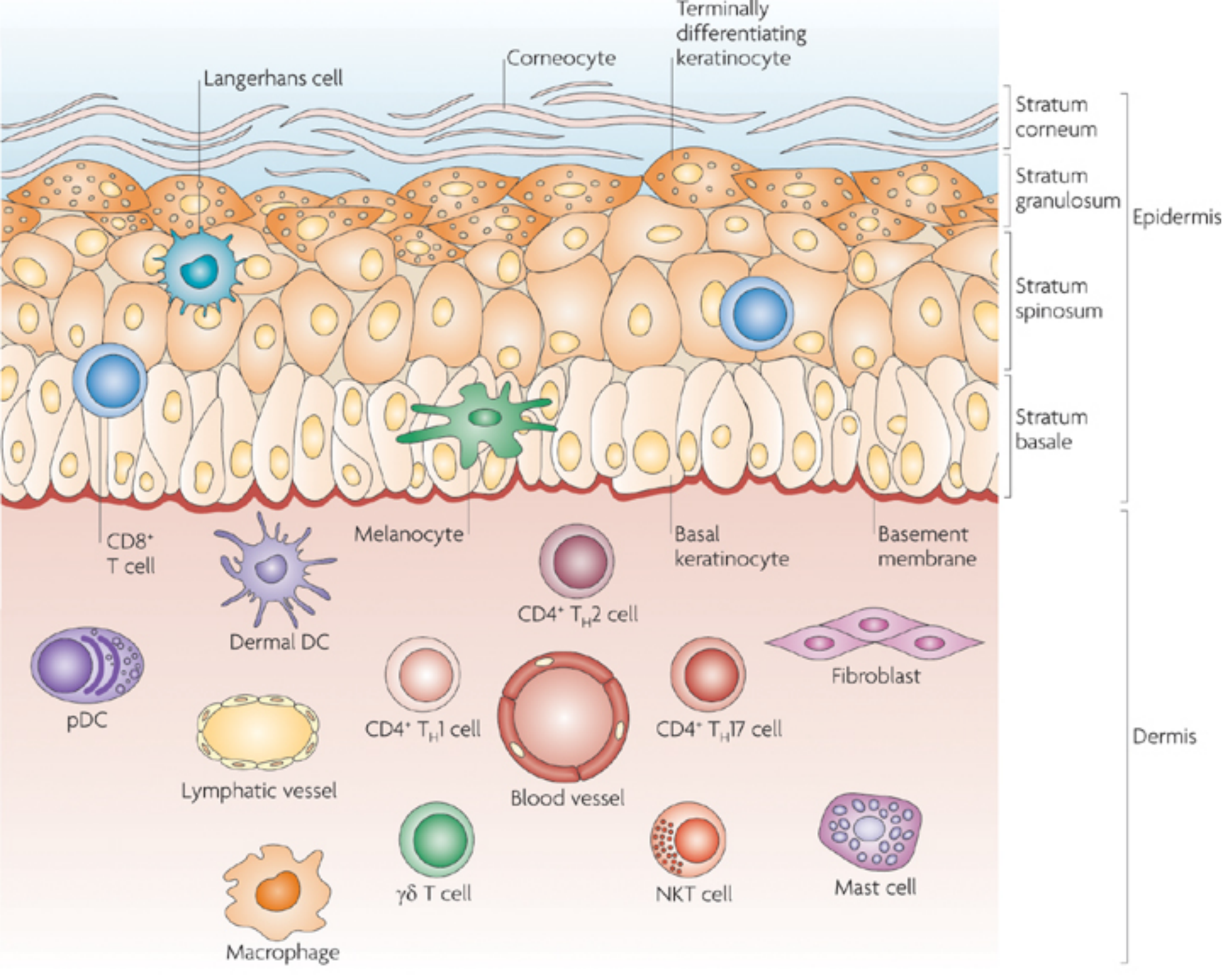
1. phuny#2s
2. fish#1ng
3. t0pph@ts
4. run$4you
5. ba#3ries

A sterile environment?

Barriers (hopefully) create a sterile internal environment

- Once you're in, you're in.

Evolution thinks that's silly.



Skin immune sentinels in health and disease
Frank O. Nestle, Paola Di Meglio, Jian-Zhong Qin & Brian J. Nickoloff
Nature Reviews Immunology 9, 679-691 (October 2009)

0x02: Escaping the Bubble

Data can make **security**
more effective and **less painful**.

If we do it right.

Requirements

Data collection from individual components + data integration

- Common data models

Security components with APIs

- …that don’t suck

Cross-organizational buy-in

- Dev, SecEng, SecOps, Network Infrastructure, etc.

Willpower

We need to engineer
an immune system.

Let’s philosophize for a second:

Security principles

Principle of Least Privilege

Least (manageable) **Privilege**

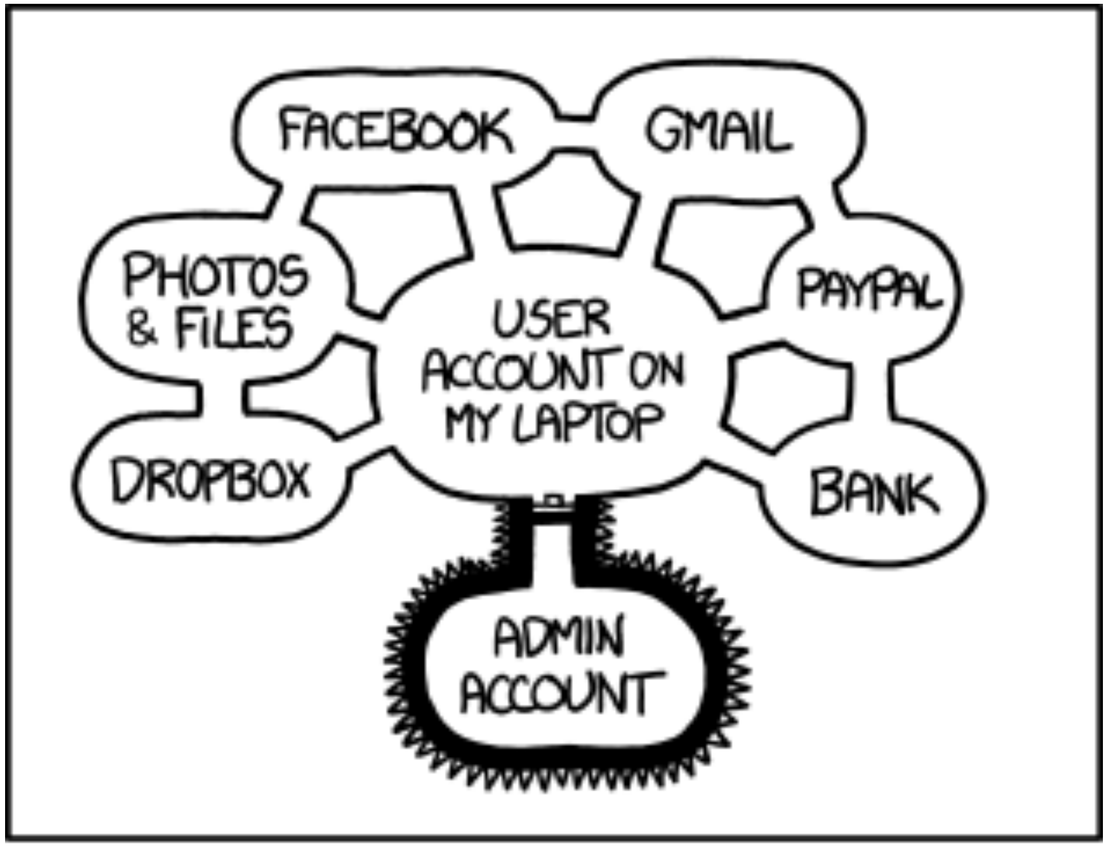
Too complex to manually configure

- either calcifies or not *really* least privilege

“Least privilege” is a dynamic concept

- users and entities don’t need all privileges all the time

Absolute Trust
Will Be Hacked Absolutely.



IF SOMEONE STEALS MY LAPTOP WHILE I'M LOGGED IN, THEY CAN READ MY EMAIL, TAKE MY MONEY, AND IMPERSONATE ME TO MY FRIENDS,

BUT AT LEAST THEY CAN'T INSTALL DRIVERS WITHOUT MY PERMISSION.

http://imgs.xkcd.com/comics/authorization.png

Bubble Exodus

Immune system = operationalized data + automation

…but many things weren’t built with this architecture in mind

- Authentication + authorization
- Endpoint security
- Network security
- Legacy applications
- etc.

Flickr user: smemon

Many are on the same path.

Google BeyondCorp

- Kill the firewall with a new proxy + authC + authZ + behavior monitoring architecture

Facebook, Square user authorization for internal resources

- Push entitlement decisions to users and managers

Facebook, Square alert response

- Users and managers handle some alerts, not SecOps

Lookout analyzing device risk to grant access to enterprise data

- Devices can access data while they're in compliance, but not when they're compromised

Micro-perimeter configuration tools

- Adaptive learning, not rigid firewall rules

0x03: Building the Immune System

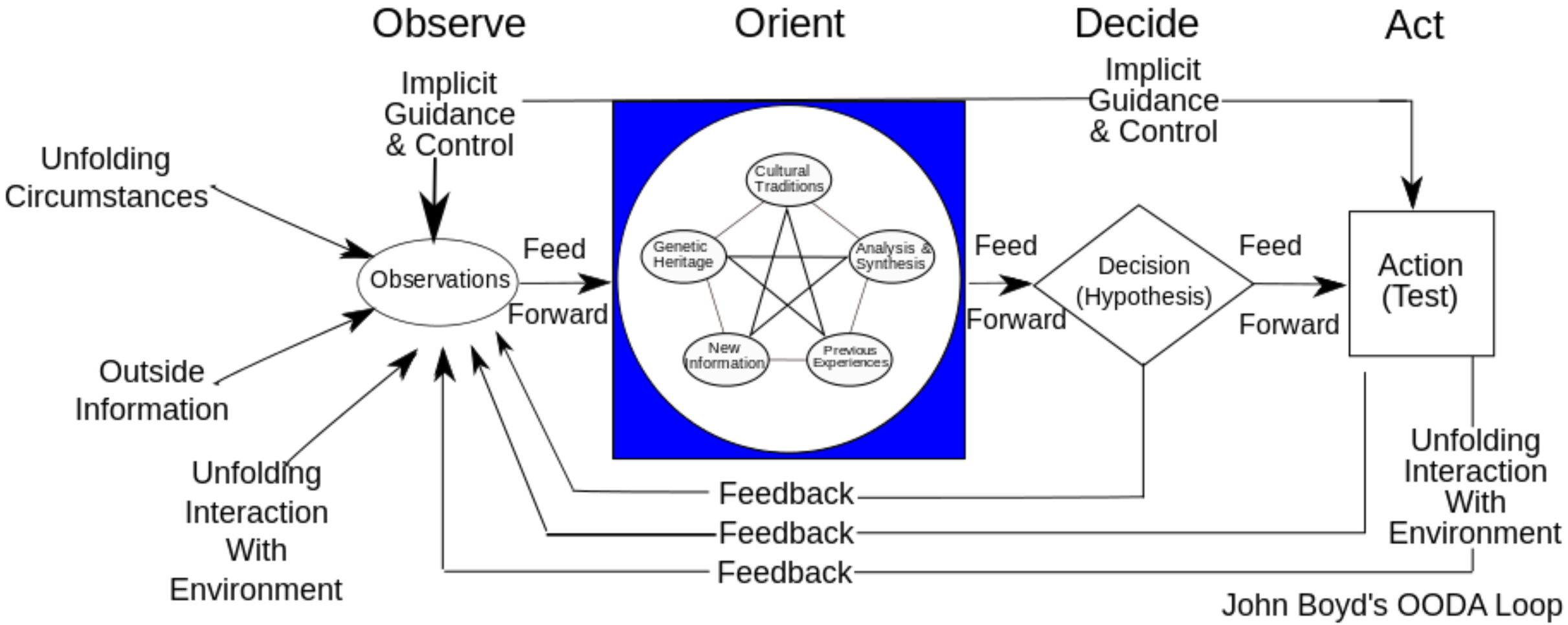
Goals

1. Decrease risk to the organization
   - by using an “immune system” to detect abuse
2. Reduce load on security team
   - Especially in configuration, investigation, and response
3. Reduce restrictions to improve UX
   - Implies that some bad things will get in that our immune system should catch

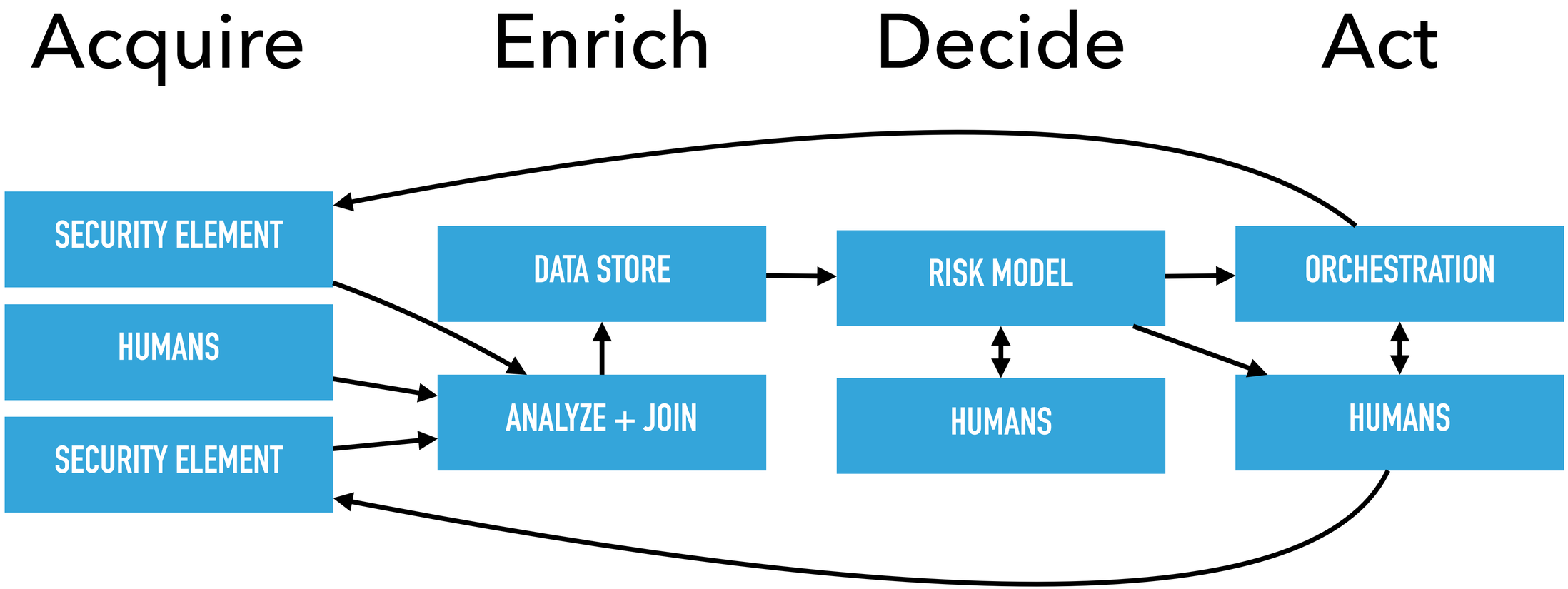
Step 0

Architect data-driven feedback loops

OODA Loop



AEDA Loop


Acquire
Enrich
Decide
Act
SECURITY ELEMENT
HUMANS
SECURITY ELEMENT
DATA STORE
ANALYZE + JOIN
RISK MODEL
HUMANS
ORCHESTRATION
HUMANS

Optimize for angular velocity

Without operational data + automation, speed is limited.

- Overworked security team
- Coarse grained policy
- Rear-view mirror security
- Dropping alerts on the floor

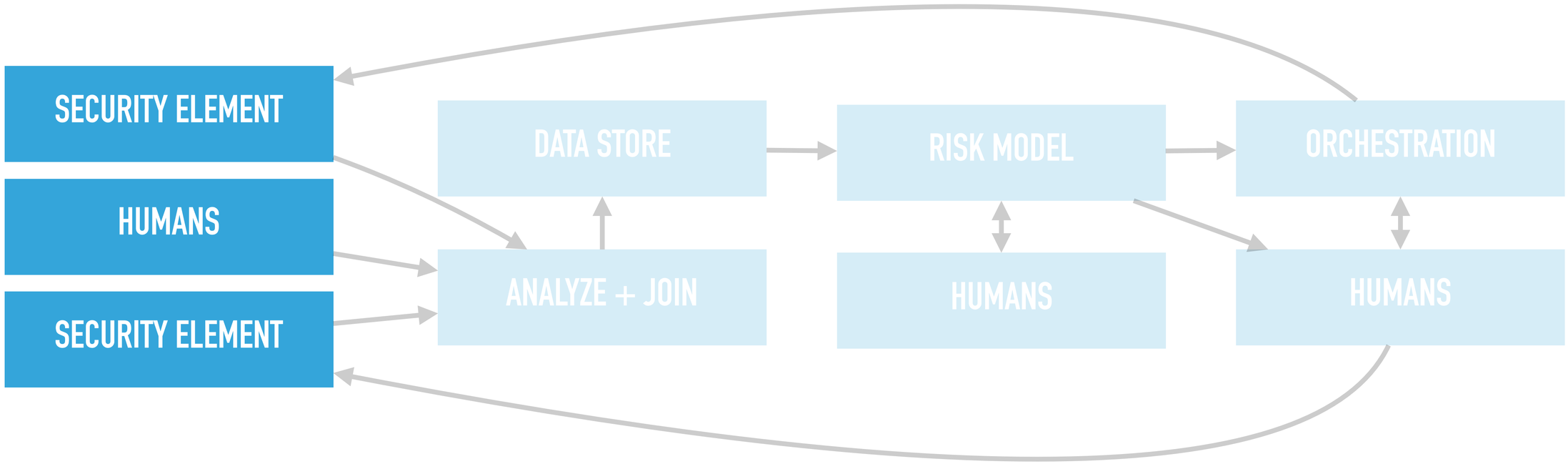
...and ultimately attackers win.

If you can iterate your loop
faster than your adversary, you win.

Step 1

Acquire

Acquire Enrich Decide Act



Test:

If [x] were compromised,
what **specific data**
would **clearly indicate** it?

I’ve never heard anyone say they have *too* much visibility into their infrastructure.

WARNING

UBIQUITOUS SURVEILLANCE

Source: https://www.flickr.com/photos/arenamontanus

Getting data from devices

Assume everything is compromised, audit all the things

- End-user devices: desktops, laptops, tablets, smartphones
- Private cloud/IaaS/PaaS: VMs, containers
- Physical servers
- SaaS applications
- Network infrastructure: routers, switches, firewalls, IPS, etc.
- Embedded devices: printers, projectors, etc.

Streaming > Batch

APIs for black-box systems

Security sensors (open source or commercial)

- Google BeyondCorp proxy infrastructure
- Lookout Mobile Threat Protection
- Facebook SSH patch to record user behavior
- Facebook OSQuery

Device or Network?

Network sensors gather netflow, application data in motion

- What is device/user A talking to?

On-device sensors gather application, OS, configuration data

- Manifests
- Behavior

Real-world deployment

On-device sensors have a compromise race condition

- Malware privilege escalates and disables sensor.
- Real-time streaming can solve this problem a lot of the time.

Network sensors increasingly unable to get data

- Mobile + cloud + remote workers == you don’t get packets with legacy infra
- Encryption == you can’t read data (unless you have an SSL terminating proxy)
- Sophisticated attackers == bad packets look like good

Build a portfolio of sensors.
All have strengths + weaknesses.

Getting data from humans

"…the **best security feature on Facebook** is something we don't have to do;
it's **the reporting mechanism** we provide for the people who use Facebook.
…
It's not just that you are a fake user and you send an inordinate number of friend requests to a category of users.
You actually also set off alarms to other people."

Joe Sullivan

http://www.csoonline.com/article/2131106/social-networking-security/securing-facebook--with-a-little-help-from-his-800-million-friends.html

The Privilege Accretion Problem

Alice get permissions to do X; she transfers to another team.

Common scenarios:

1. The company doesn't change her privileges upon moving.

2. Alice is so good, her former team asks to have her back two weeks per year, so she keep her privileges.

PRODUCTION PORTAL (SJC1B)

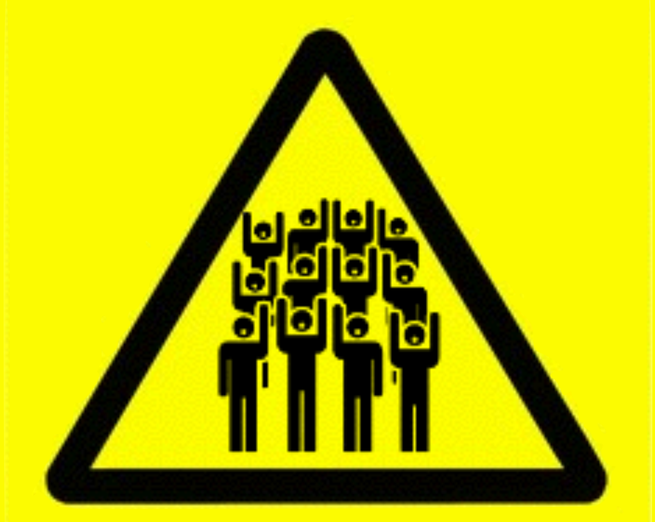
LOGGED IN AS DIOGO • LOGOUT

Pending Access Requests

Username	Application	Capability	Created	Action
	Franklin	admin	about 5 hours ago	Approve Deny
	Admin Dashboard	admin	about 6 hours ago	Approve Deny
	Admin Dashboard	admin	about 7 hours ago	Approve Deny
	Franklin	admin	20 days ago	Approve Deny
	Admin Dashboard	admin	25 days ago	Approve Deny
	Admin Dashboard	admin	25 days ago	Approve Deny
	Regulator	common_user	about 1 month ago	Approve Deny
	Admin Dashboard	admin	about 2 months ago	Approve Deny
	Admin Dashboard	admin	3 months ago	Approve Deny
	Admin Dashboard	admin	3 months ago	Approve Deny
	Admin Dashboard	admin	3 months ago	Approve Deny

Stolen from Diogo Mónica's Security@Scale talk

WARNING



GROUP INTELLECT

Source: https://www.flickr.com/photos/arenamontanus

Some humans are lazy and unreliable

Some managers allow every request

- No incentive for manager to say anything other than yes.

Solution: don’t ask about privileges, ask about roles.

- Role-modeling: person has many roles, role has many privileges
- Use statistics to transform privileges to roles.

What if you need access right away?

You don’t want to grant lots of rarely-used privileges

- e.g. SSH access to servers for developers

Solution: Emergency “break glass” access

- Alerts manager, security team, but allows the business to move on.

Data storage

Regulatory Issues

Is your organization subject to regulations that would limit data collection and storage?

Avoid geo-isolating data however you can!

- Don’t collect sensitive data
- Scrub sensitive data
- Anonymize

Data Infrastructure

Immutable logs

- e.g. Kafka

“Big data” systems

- e.g. Hadoop, Spark

Next-generation DBs

- e.g. Cassandra, Elastic Search, CouchDB, CockroachDB, Riak

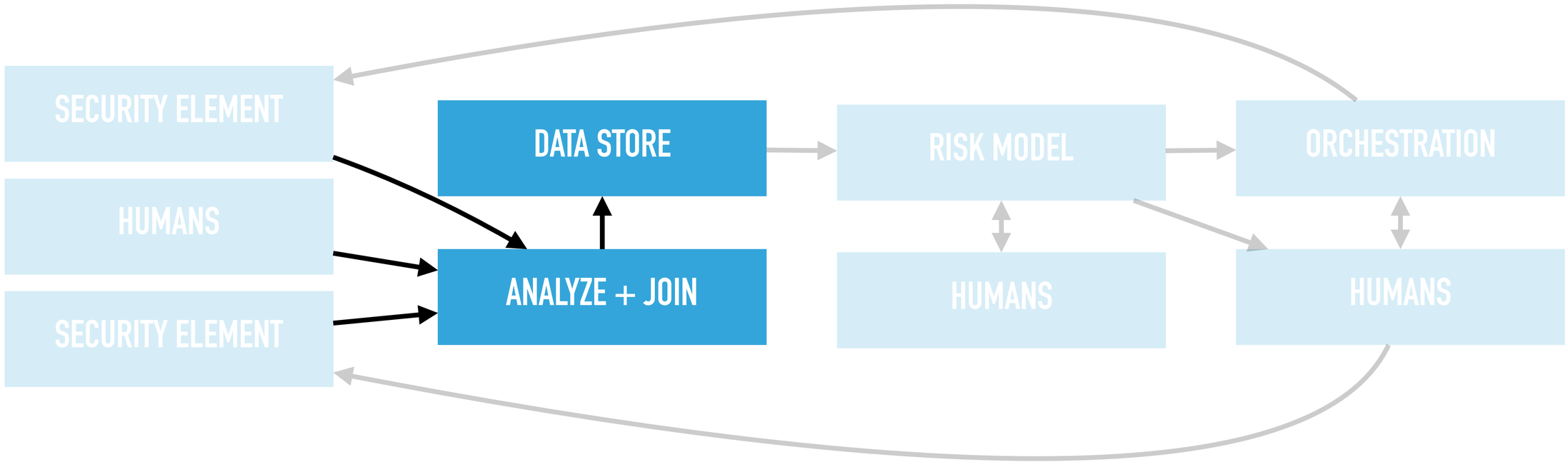
Security-specific infrastructure

- e.g. SIEM, UEBA

Step 2

Enrich

Acquire Enrich Decide Act



Problem: Many systems don’t have enough data to be effective.

Choose one:

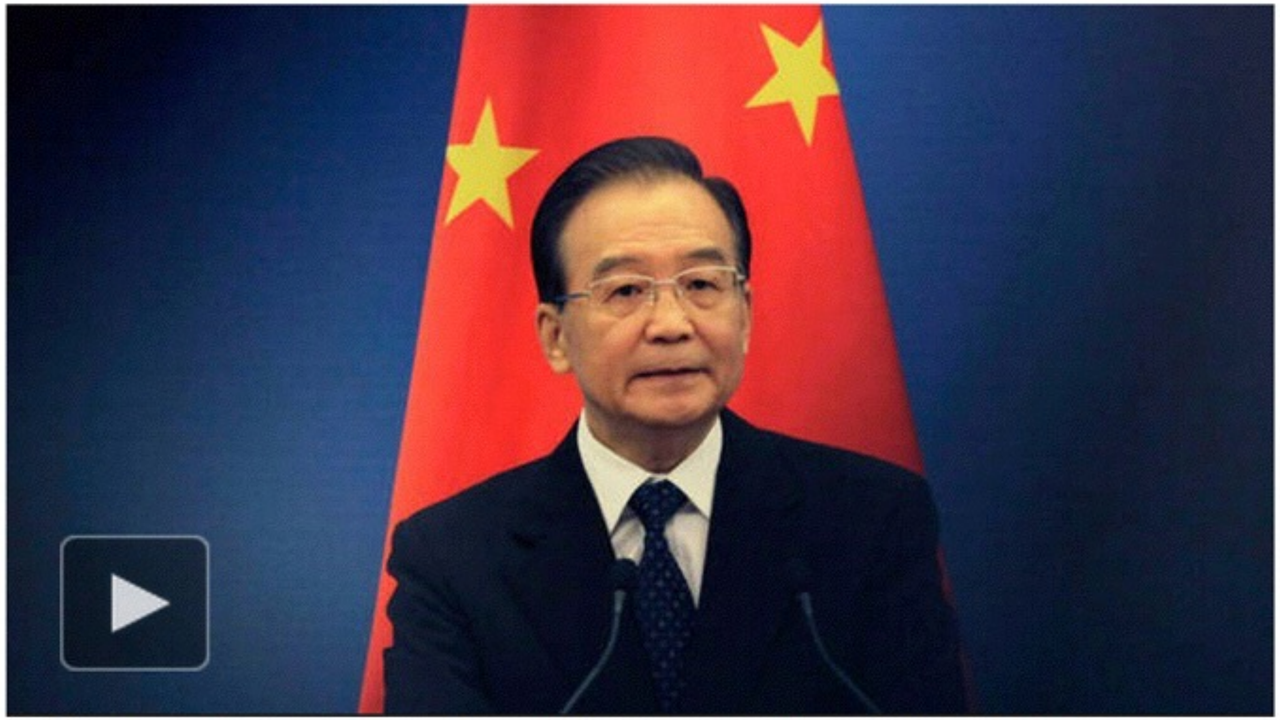
too many **false positives**

–or–

too many **false negatives**

Systems that **minimize false negatives** tend to **increase false positives**.

Hackers in China Attacked The Times for Last 4 Months



A Cyberattack From China: TimesCast: Chinese hackers infiltrated The New York Times's computer systems, getting passwords for its reporters and others.

By NICOLE PERLROTH
Published: January 30, 2013 | 391 Comments

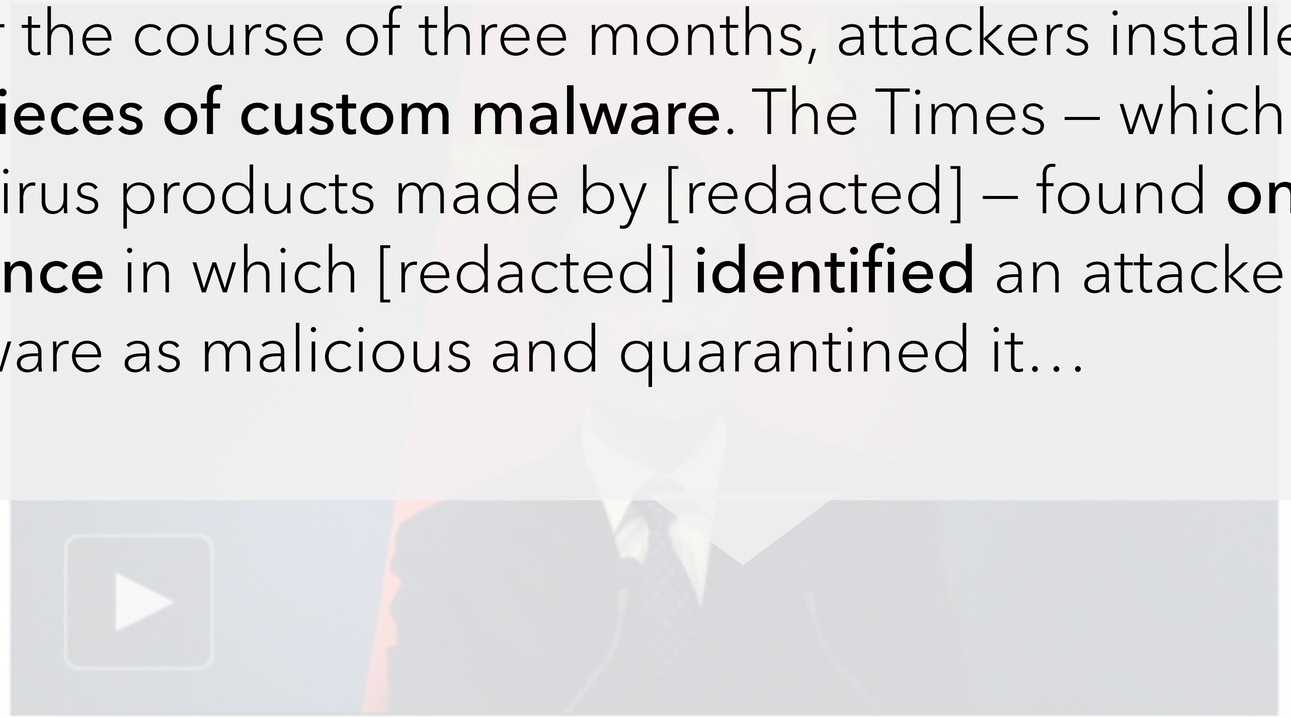
HOME PAGE | TODAY'S PAPER | VIDEO | MOST POPULAR | U.S. Edition

The New York Times

Business Day
Technology

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH

Hackers in China Attacked The Times for Last 4 Months



A Cyberattack From China: TimesCast: Chinese hackers infiltrated The New York Times's computer systems, getting passwords for its reporters and others.

By NICOLE PERLROTH
Published: January 30, 2013 | 391 Comments

“

Over the course of three months, attackers installed **45 pieces of custom malware**. The Times – which uses antivirus products made by [redacted] – found **only one instance** in which [redacted] **identified** an attacker’s software as malicious and quarantined it…

Hackers in China Attacked The Times for Last 4 Months

“

Over the course of three months, attackers installed **45 pieces of custom malware**. The Times – which uses antivirus products made by [redacted] – found **only one instance** in which [redacted] **identified** an attacker’s software as malicious and quarantined it…

By NICOLE PERLROTH

Systems that **minimize false positives** tend to **increase false negatives**.

For a given set of input data, there's a **fundamental tradeoff** between **false positives** and **false negatives.**

For a given set of input data, there's a **fundamental tradeoff** between **false positives** and **false negatives.**

To reach **greater efficiency**,
your data needs **more context**.

Problem: Many systems have steaming piles of non-operational data

u@server:~$ **cat /log/access.log | grep 23.11.541.66**

Records: 100 / 4154 1486 ms < prev **1** 2 3 4 5 6 7 next > 15

	Timestamp	Fields
Info	Tue Nov 22 08:53:20	**1321973538.778549|vfLpkUrpoI6|10.124.19.12|47263|209.85.225.132|443|TLSv10|TLS_ECDHE_RSA_WITH_RC4_128_SHA|s2.goo View,ST=California,C=US|1320932962.000000|1352555962.000000|0ef6837e26d26f08700a9e03c863dafe|ok** host=165.189.226.172 program=bro_ssl class=BRO_SSL srcip=10.124.19.12 srcport=47263 dstip=209.85.225.132 dstport=443 expira View,ST=California,C=US
Info	Tue Nov 22 08:53:20	**1321973537.891299|oE6L8vilUv7|10.124.19.12|41018|199.59.149.198|443|TLSv10|TLS_RSA_WITH_RC4_128_SHA|twitter.com|97 Inc.,streetAddress=795 Folsom St\, Suite 600,L=San Francisco,ST=California,postalCode=94107,C=US,serialNumber=4337446,2.5.4.15=#131450726976617465204F7267616E697A61** host=165.189.226.172 program=bro_ssl class=BRO_SSL srcip=10.124.19.12 srcport=41018 dstip=199.59.149.198 dstport=443 expira Folsom St, Suite 600,L=San Francisco,ST=California,postalCode=94107,C=US,serialNumber=4337446,2.5.4.15=#1314507269766174
Info	Tue Nov 22 08:53:25	**Teardown UDP connection 144744478313156395 for DET-SEC-124.19:10.124.19.12/45091 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=4509
Info	Tue Nov 22 08:53:25	**Teardown UDP connection 144744478313156396 for DET-SEC-124.19:10.124.19.12/52757 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=5275
Info	Tue Nov 22 08:53:26	**Teardown UDP connection 144744478313156397 for DET-SEC-124.19:10.124.19.12/47309 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=4730
Info	Tue Nov 22 08:53:26	**Teardown UDP connection 144744478313156398 for DET-SEC-124.19:10.124.19.12/52485 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=5248
Info	Tue Nov 22 08:53:26	**Teardown UDP connection 144744478313156399 for DET-SEC-124.19:10.124.19.12/57404 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=5740
Info	Tue Nov 22 08:54:20	**Teardown UDP connection 144744478313156408 for DET-SEC-124.19:10.124.19.12/35728 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=3572
Info	Tue Nov 22 08:54:20	**Teardown UDP connection 144744478313156409 for DET-SEC-124.19:10.124.19.12/43103 to OUTSIDE:10.68.15.11/53 duration 0** host=165.189.82.68 program=%fwsm-5-302016 class=FIREWALL_CONNECTION_END proto=UDP srcip=10.124.19.12 srcport=4310
Info	Tue Nov 22	**Teardown UDP connection 144744478313156410 for DET-SEC-124.19:10.124.19.12/51752 to OUTSIDE:10.68.15.11/53 duration 0**

HINDSIGHT

Those really were the droids you were looking for.

Two Techniques:

1. Analyzing
2. Joining

Analyze data

Data Information Knowledge Wisdom

understanding

Turn data into information

Static/Dynamic analysis of executables, content

- `strings datafile`
- Behavioral analysis sandboxes
- Control flow analysis

Parsing of protocols, files

- Deep packet inspection
- Structure-based exploits

Data normalization

- Canonicalization (e.g. phone numbers, character encodings)

You **can’t extract information**
that’s **unsupported**
by the **underlying data**.

Joining datasets

Problem: isolated data is of limited value

- MAC address 00:00:DE:AD:BE:EF sent 1000 packets to 00:08:BA:AD:F0:0D

Increase information by linking datasets together

- Alice in accounting's tablet sent 1000 packets to Boston Engineering's git repo

Foreign Keys

Can’t join datasets that don’t have >= 1 factor to correlate.

- Common keys: user account id, device id, IP, domain, MAC address

Need to normalize data to ensure join is smooth

- Unicode encodings
- SHA-1 vs. MD5 vs. SHA-256

Scope of data joins

System-internal

Organization-wide

External

- Industry/National
- Global

Organization-wide example

HR vacation data + source code commit data

If you commit code while on vacation, you get an email.


PANIC

External data

Data from outside the building, e.g.

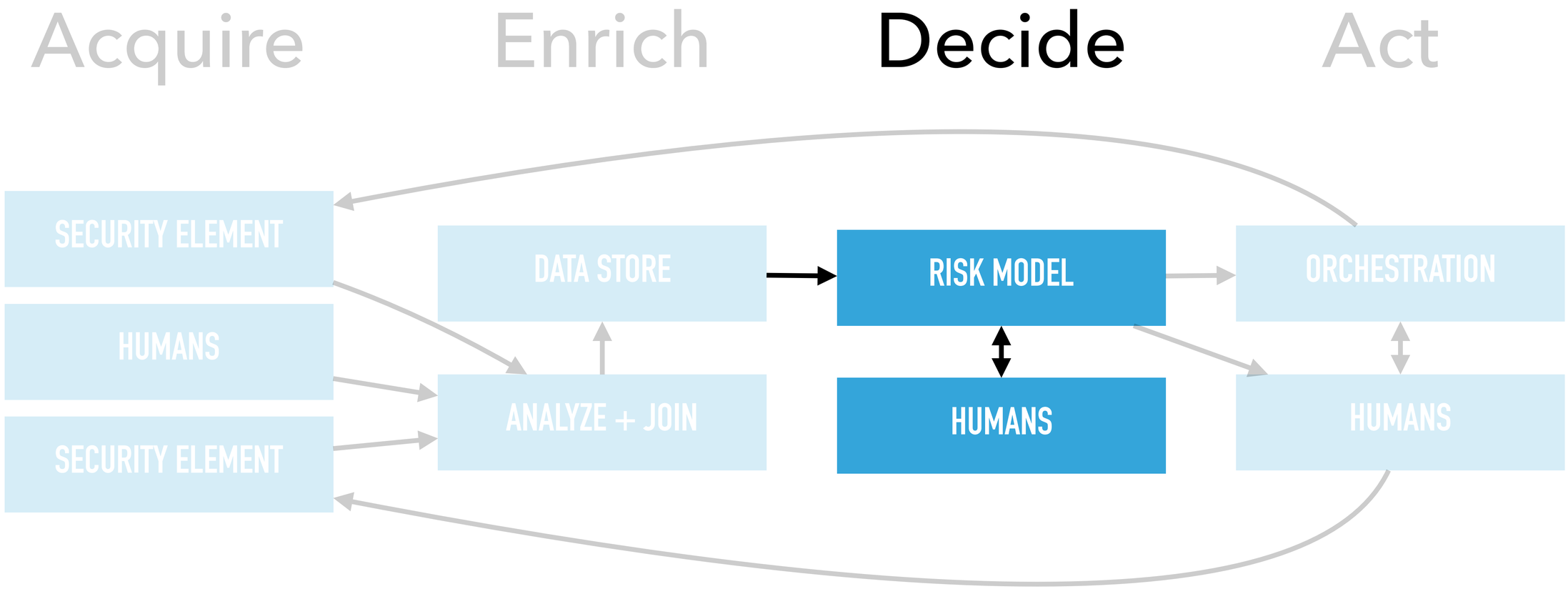
- GeoIP
- IP, domain reputation
- Known malware

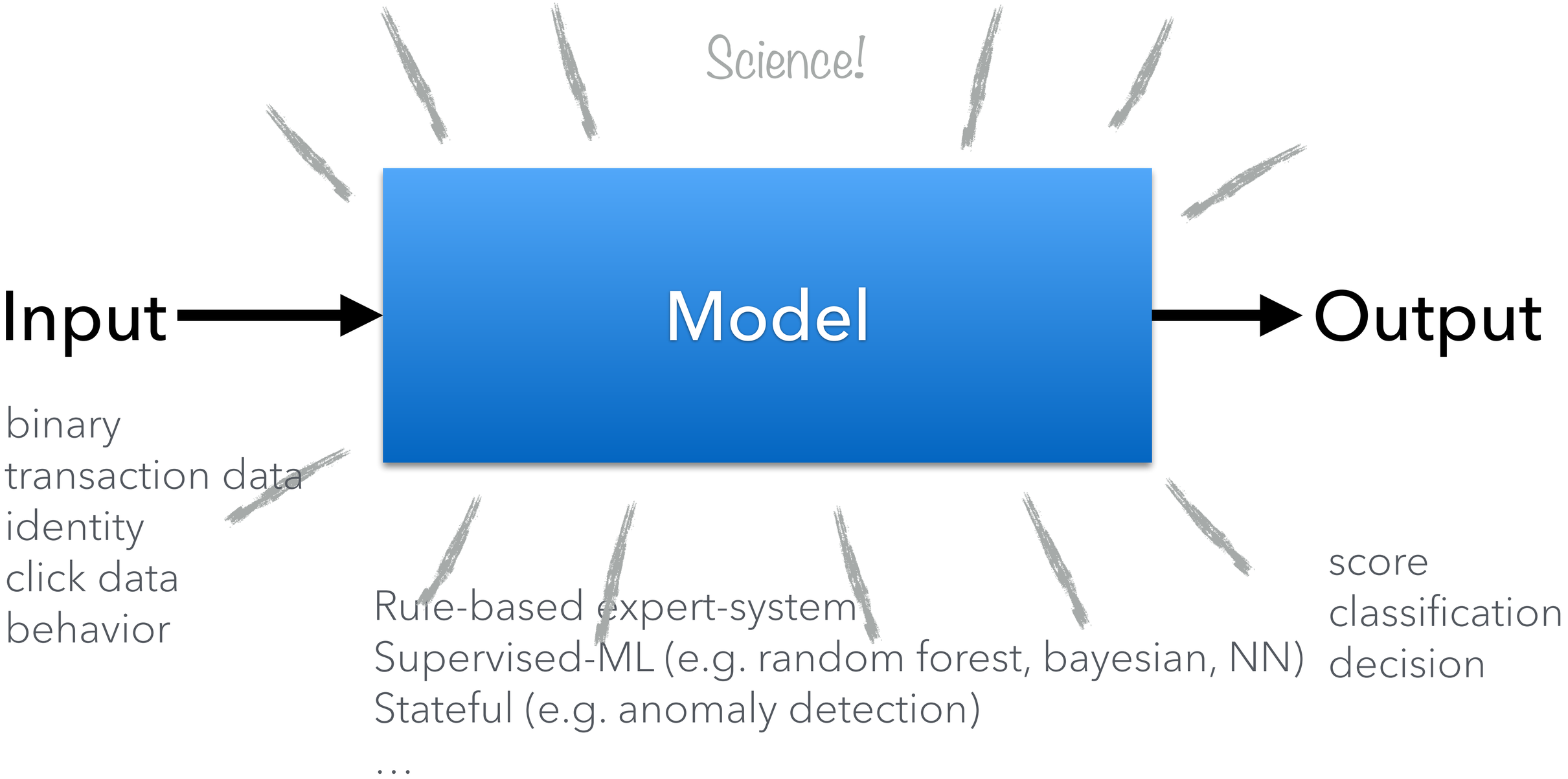
How do you handle conflicts between data sources?

Do you trust the data?

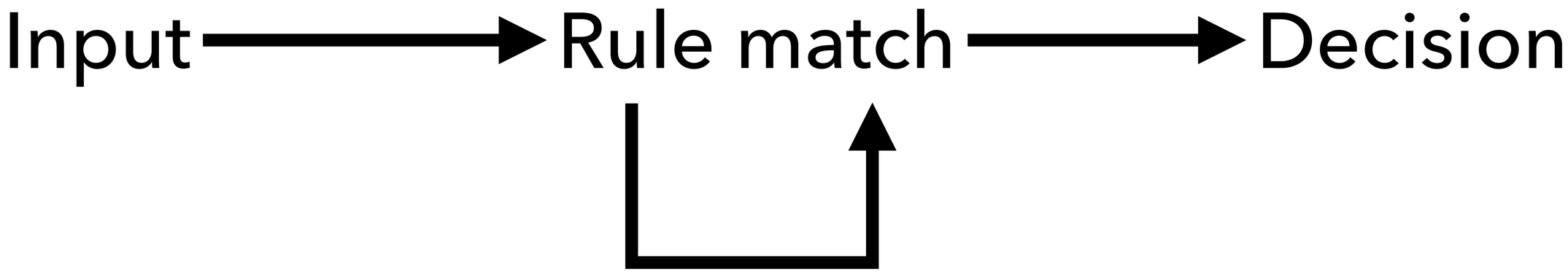
Step 3

Decide


Acquire
Enrich
Decide
Act
SECURITY ELEMENT
HUMANS
SECURITY ELEMENT
DATA STORE
ANALYZE + JOIN
RISK MODEL
HUMANS
ORCHESTRATION
HUMANS


Science!
Input
Model
Output
binary
transaction data
identity
click data
behavior
Rule-based expert-system
Supervised-ML (e.g. random forest, bayesian, NN)
Stateful (e.g. anomaly detection)
…
score
classification
decision

Rule-based expert systems



Logical combinations of rules

Anomaly Detection

Good

Great at finding novel threats

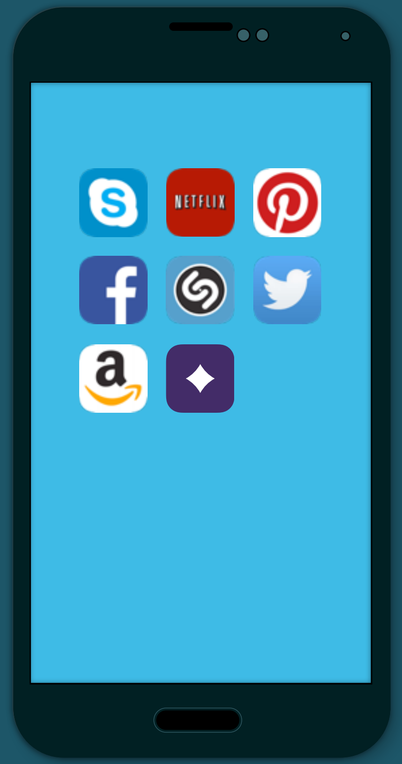
Bad

Great at keeping the SOC busy

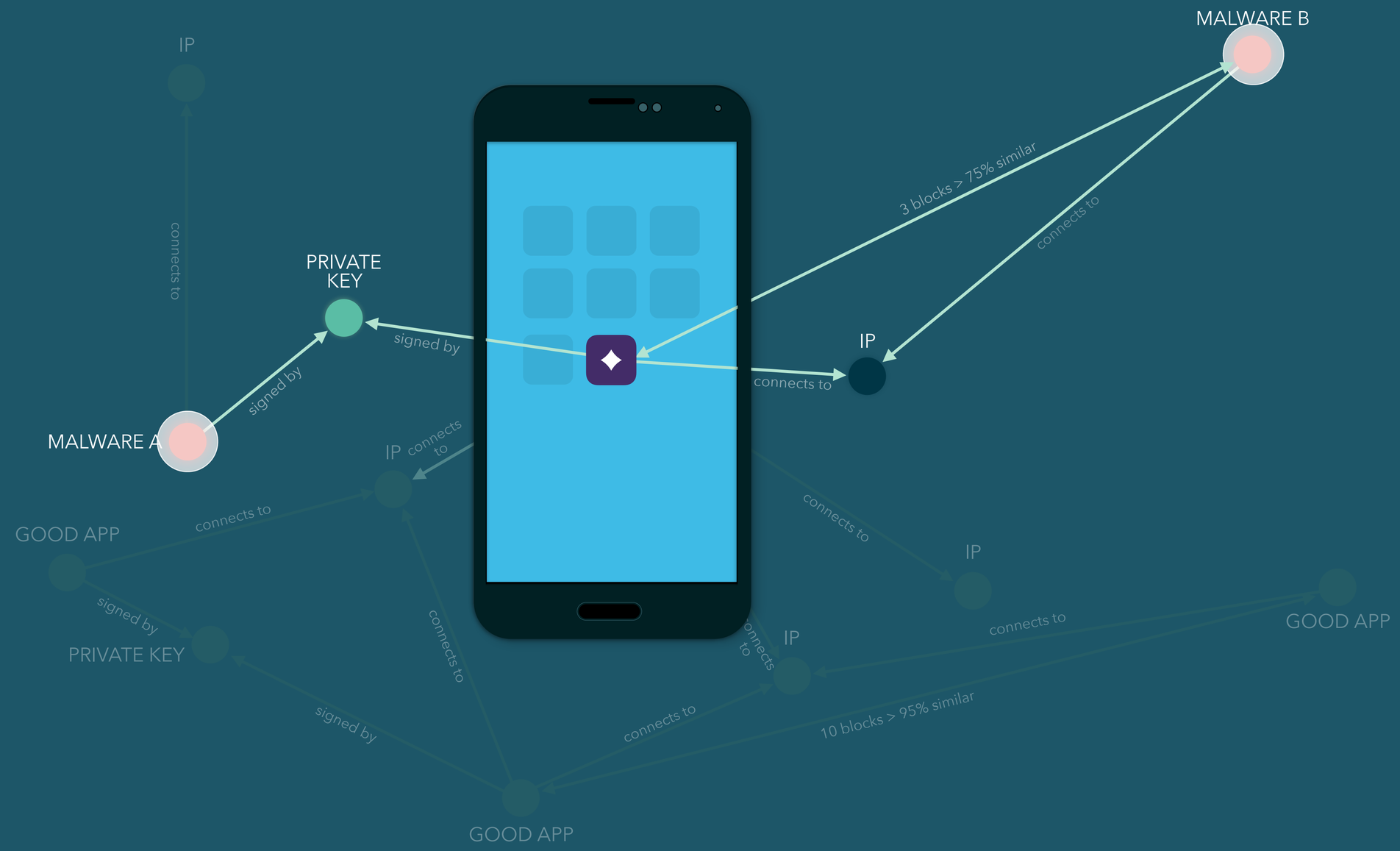
Can be effective if you have enough context.

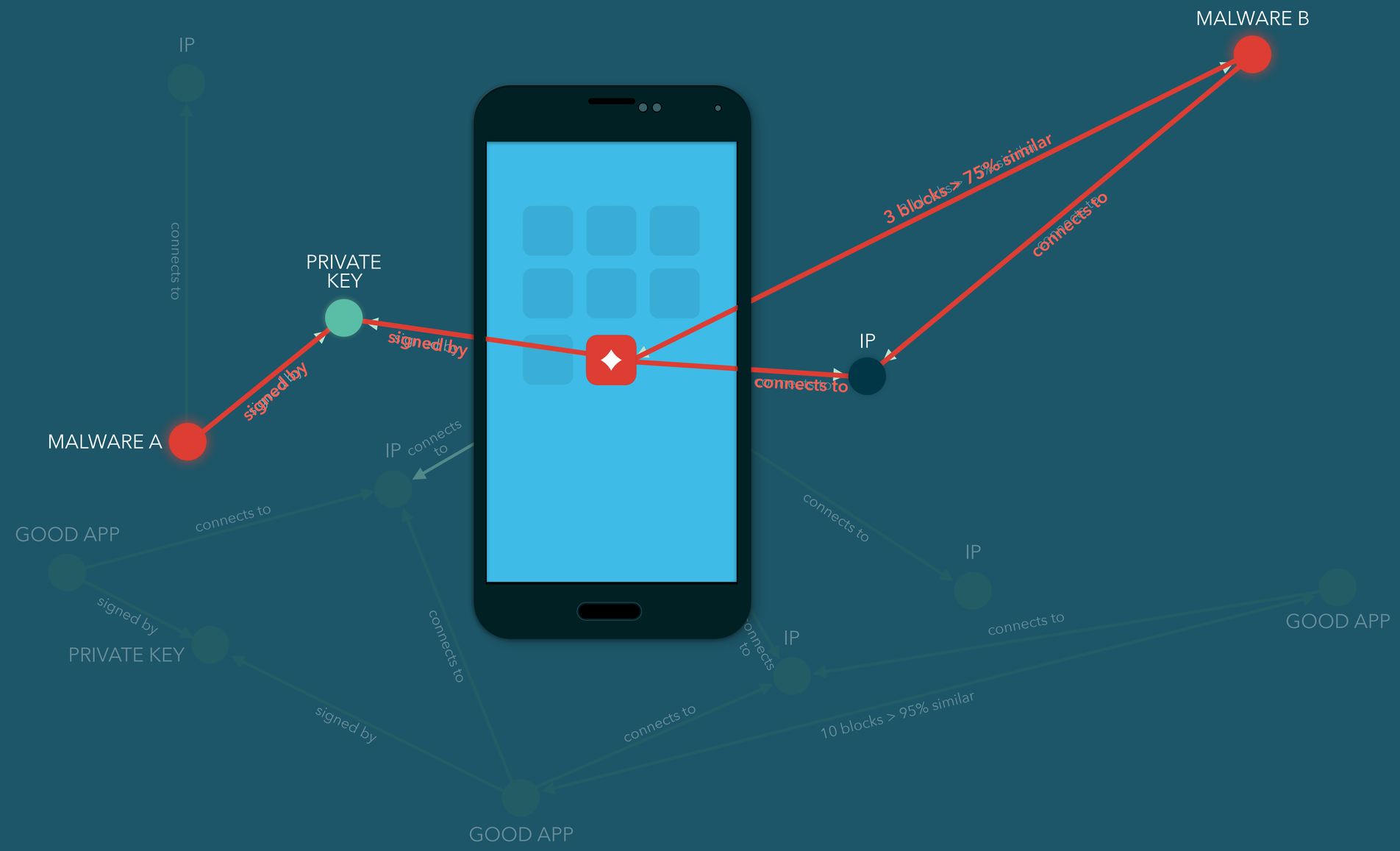
- Peer analysis: Is Bob behaving like others in his role?

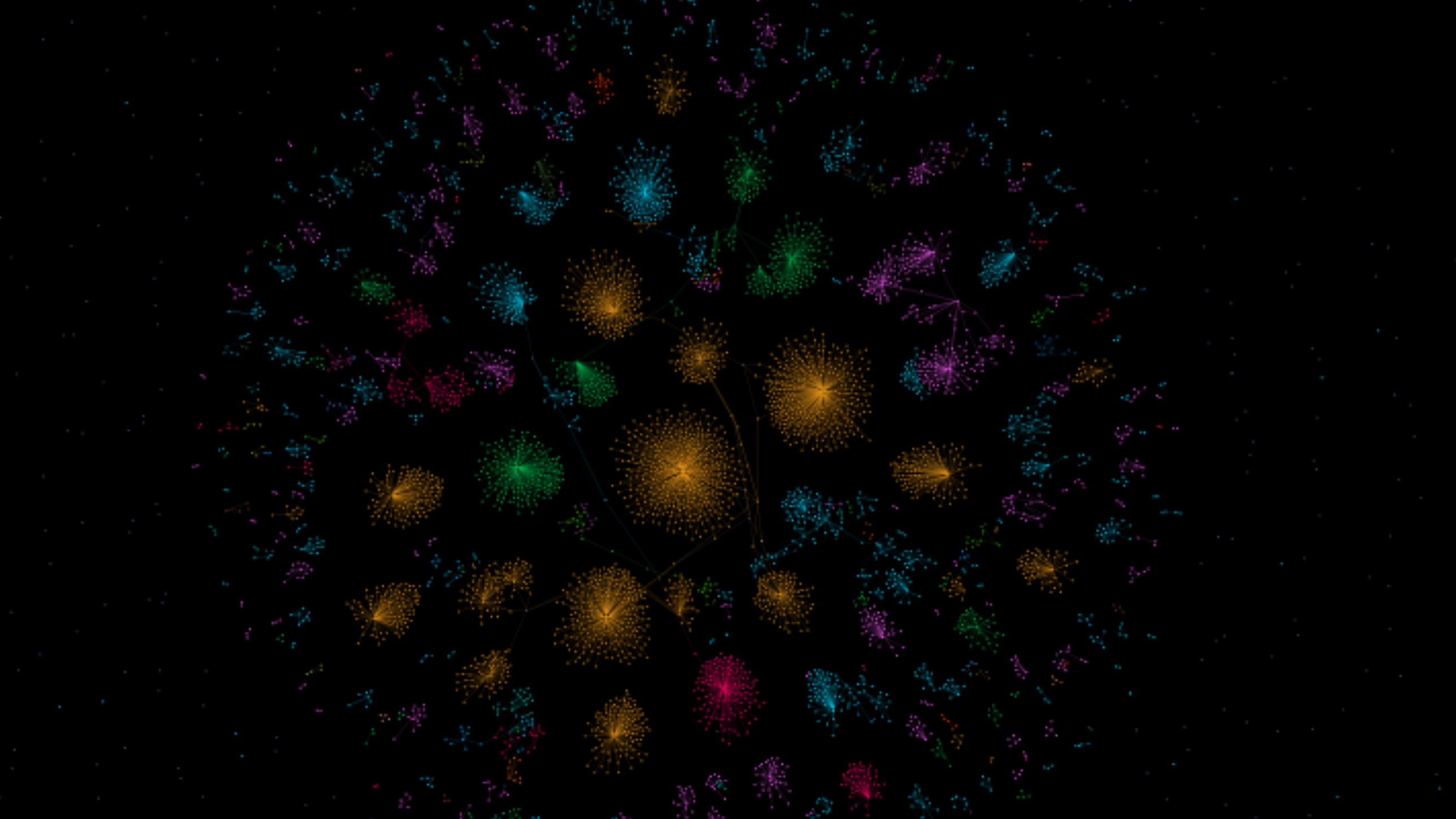
Correlation to known threats


NETFLIX


IP
connects to
PRIVATE KEY
signed by
signed by
MALWARE A
IP
connects to
GOOD APP
connects to
signed by
PRIVATE KEY
connects to
signed by
GOOD APP
3 blocks > 75% similar
connects to
IP
connects to
connects to
IP
connects to
IP
connects to
connects to
10 blocks > 95% similar
GOOD APP


IP
connects to
MALWARE B
3 blocks > 75% similar
connects to
PRIVATE KEY
signed by
IP
connects to
signed by
MALWARE A
IP
connects to
connects to
GOOD APP
connects to
IP
signed by
PRIVATE KEY
connects to
connects to
IP
connects to
GOOD APP
signed by
connects to
10 blocks > 95% similar
GOOD APP


MALWARE B
IP
connects to
PRIVATE KEY
signed by
3 blocks > 75% similar
connects to
IP
connects to
signed by
MALWARE A
IP
connects to
GOOD APP
connects to
connects to
IP
signed by
PRIVATE KEY
connects to
connects to
IP
connects to
GOOD APP
signed by
connects to
10 blocks > 95% similar
GOOD APP



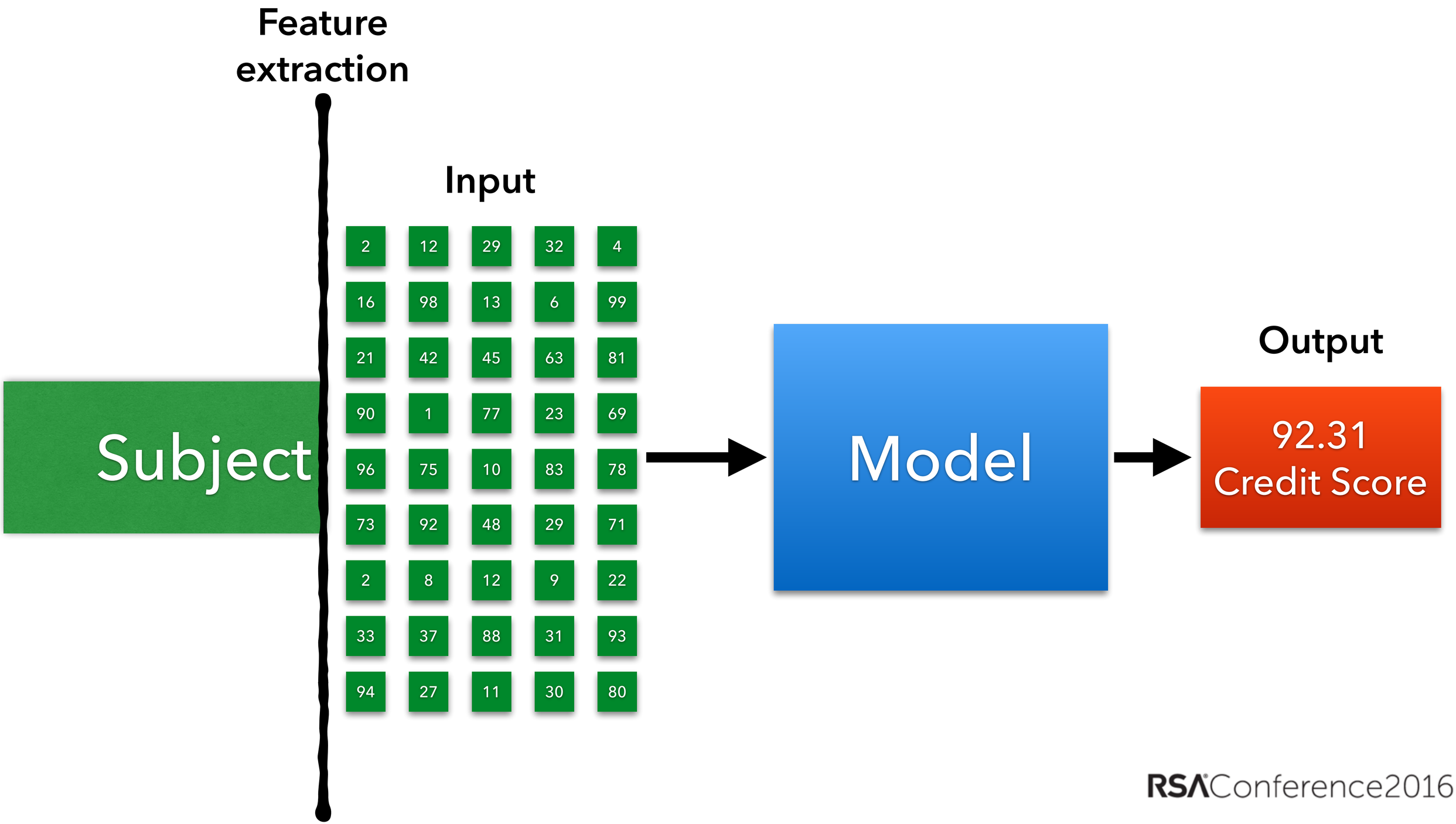
Supervised ML

Inputs: “features” of something you’re trying to make a prediction about.

Outputs: the prediction you’re trying to make.

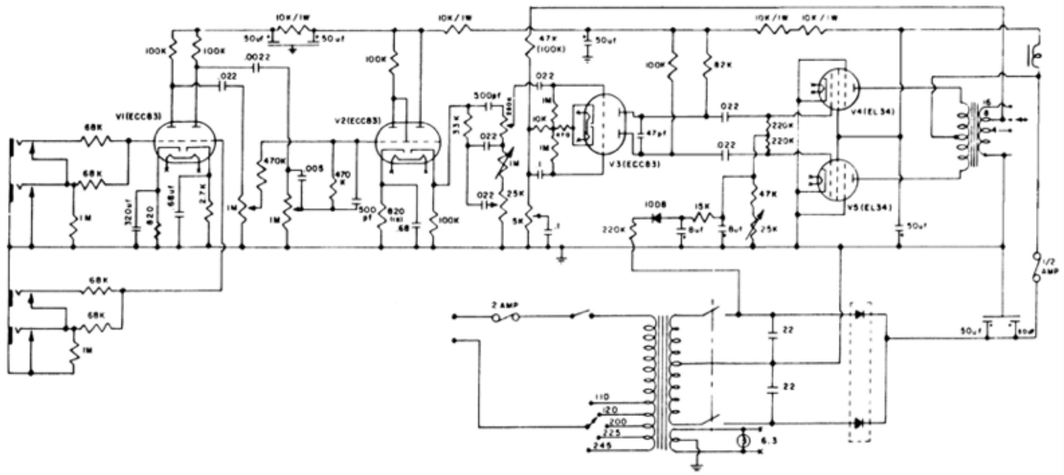
Function of model: transform some inputs to some outputs based on a model.

Training: take known inputs and outputs and make a model that transforms input to output.


Feature extraction
Input
2 12 29 32 4
16 98 13 6 99
21 42 45 63 81
90 1 77 23 69
96 75 10 83 78
73 92 48 29 71
2 8 12 9 22
33 37 88 31 93
94 27 11 30 80
Subject
Model
Output
92.31
Credit Score

Feature engineering is the art behind the science

1. Pre-requisite: Understand the field
2. Identify candidate features
3. Build tools to extract them



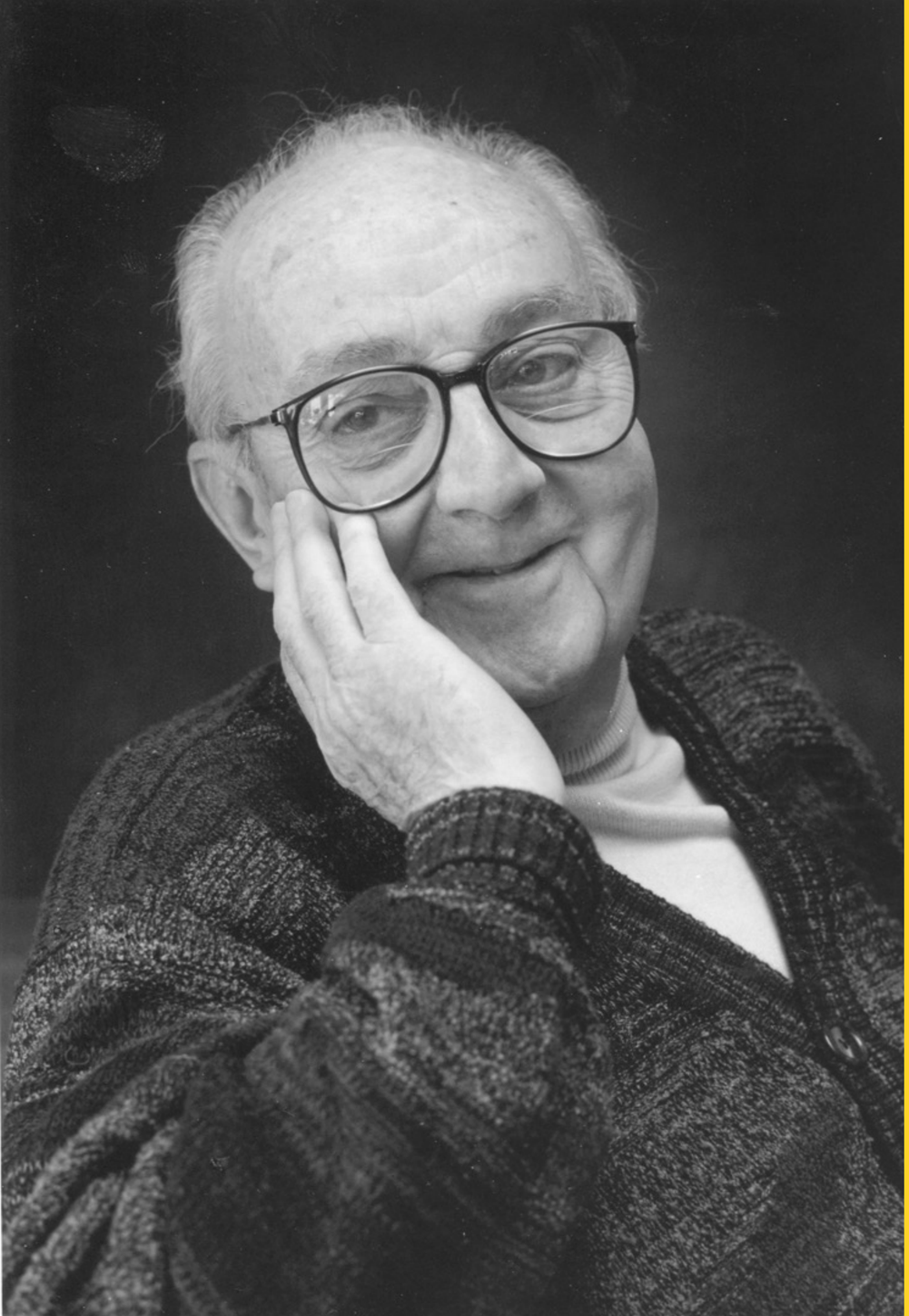
Compound Systems

Combine multiple types of systems together

Logic can combine in interesting ways

- If signature or heuristic flag an executable as bad, block it, else run behavioral analysis.

Many modern decision systems combine multiple approaches



…all models are wrong;
but some are useful.

George Box

Malware Models

Known malware

- Expert system based on known threat indicators
- Signatures!

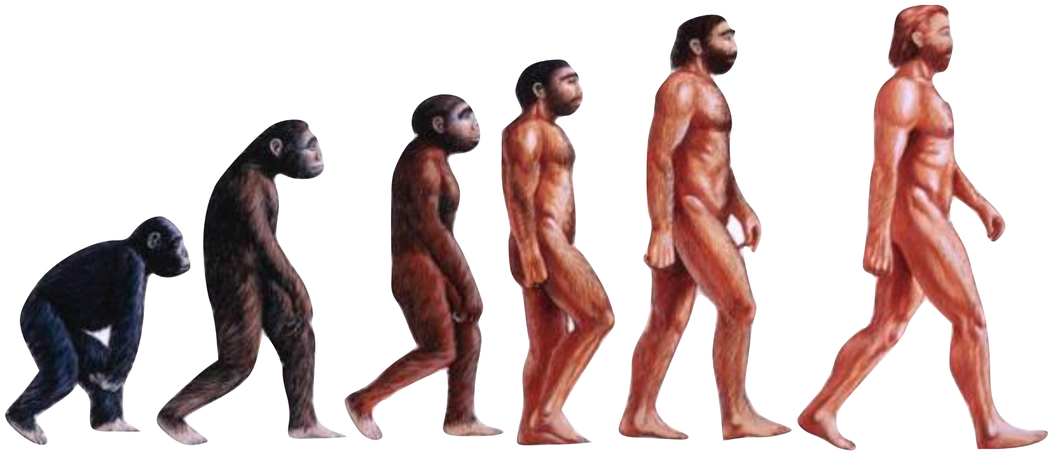
Correlated 0-day: new malware from known actor

- Traverse causal connections.
- Requires LOTS OF DATA!

Uncorrelated 0-day: new malware from unknown actor

- Supervised/unsupervised learning + hand-tuned risk models.
- Expensive and noisy!

Machines cannot (currently) make all decisions.



Hunches

Can judge “right” from “wrong”

Limited in working memory

Handle large + complex datasets

Not (yet) a general intelligence

Willing to work all night

https://www.flickr.com/photos/kosmograddotnet

Build the cyborg!

Is your data operational?

Are you asking machines questions?

- Decision support

Are machines asking you questions?

- Operational


VISA
4000 1234 5678 9010
GOOD THRU 12/20
CARDHOLDER NAME
VISA

Finding new threats with cyborgs

Anomaly detection in an environment where you *expect* new things, falls over.

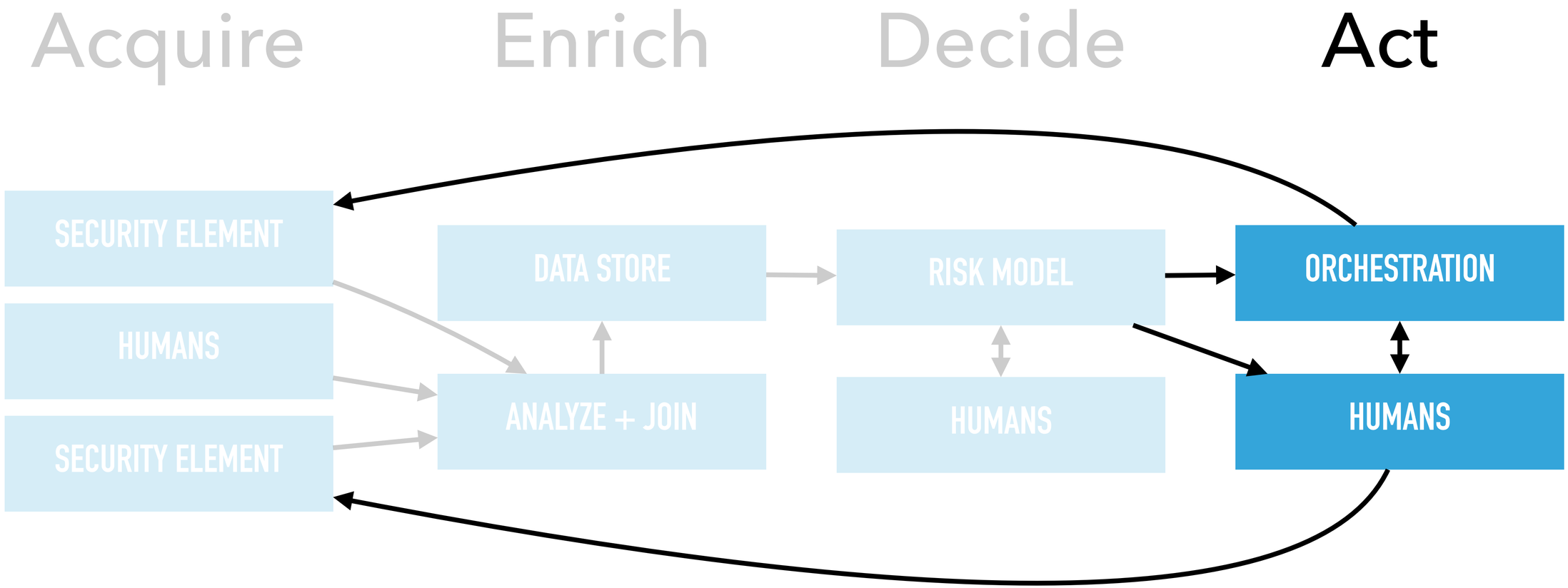
- New != bad

Expert systems: hypothesize indicators of compromise + escalate + suppress

Supervised ML: train on is_suspicious, not is_malicious

Step 4

Act


Acquire
Enrich
Decide
Act
SECURITY ELEMENT
HUMANS
SECURITY ELEMENT
DATA STORE
ANALYZE + JOIN
RISK MODEL
HUMANS
ORCHESTRATION
HUMANS

Problem: not enough humans to handle incident load.

WARNING

AUTONOMOUS DEVICE

Source: https://www.flickr.com/photos/arenamontanus

Start by improving IR UX

- Does your IR team spend hours tracking down data for each alert?

…then automate orchestration

- Does your IR team have to manually remediate systems?

…then pull the humans out

- Carefully.

IR UX

Gather all data in one place

- SIEM or custom system
- Data pipeline must have low-enough latency

Make sure the UX to interact with the data is good

- May be commercial or custom in-house

Orchestration feedback loops

Configuration

Code policy

- Block executables from running

Firewall/proxy rules

- Lock down east-west traffic
- Disallow high risk ingress/egress traffic

Content-rights

- Prevent high risk endpoint from accessing data

NAC

- Block high risk device from network

Revoke authorization to apps/data

Authentication

Force password change

- e.g. malware on device that can engage in phishing

Step-up authentication

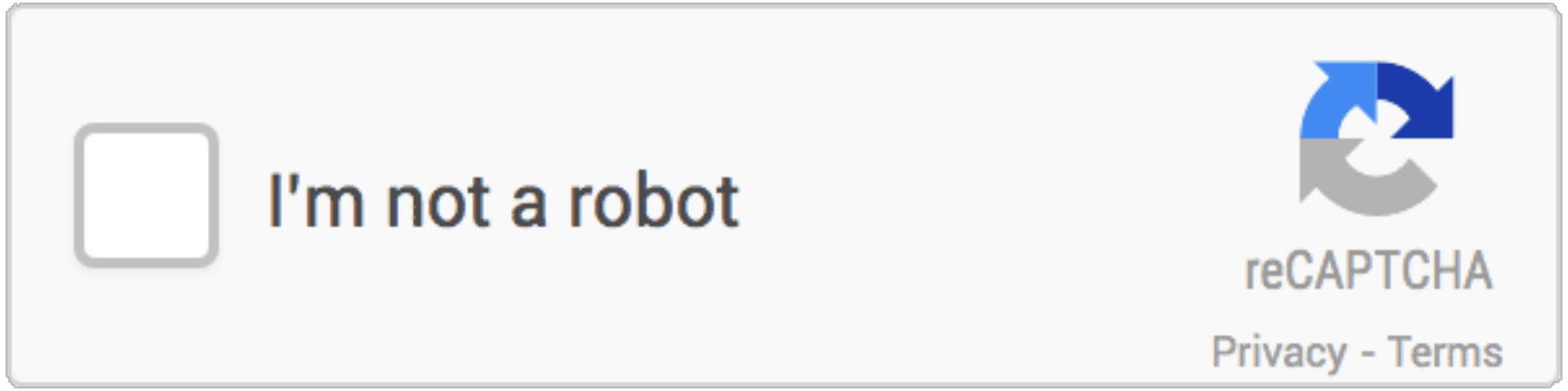
- Don’t always require MFA + short session timeouts
- Use risk to determine authentication requirements

Friend 2 of 7

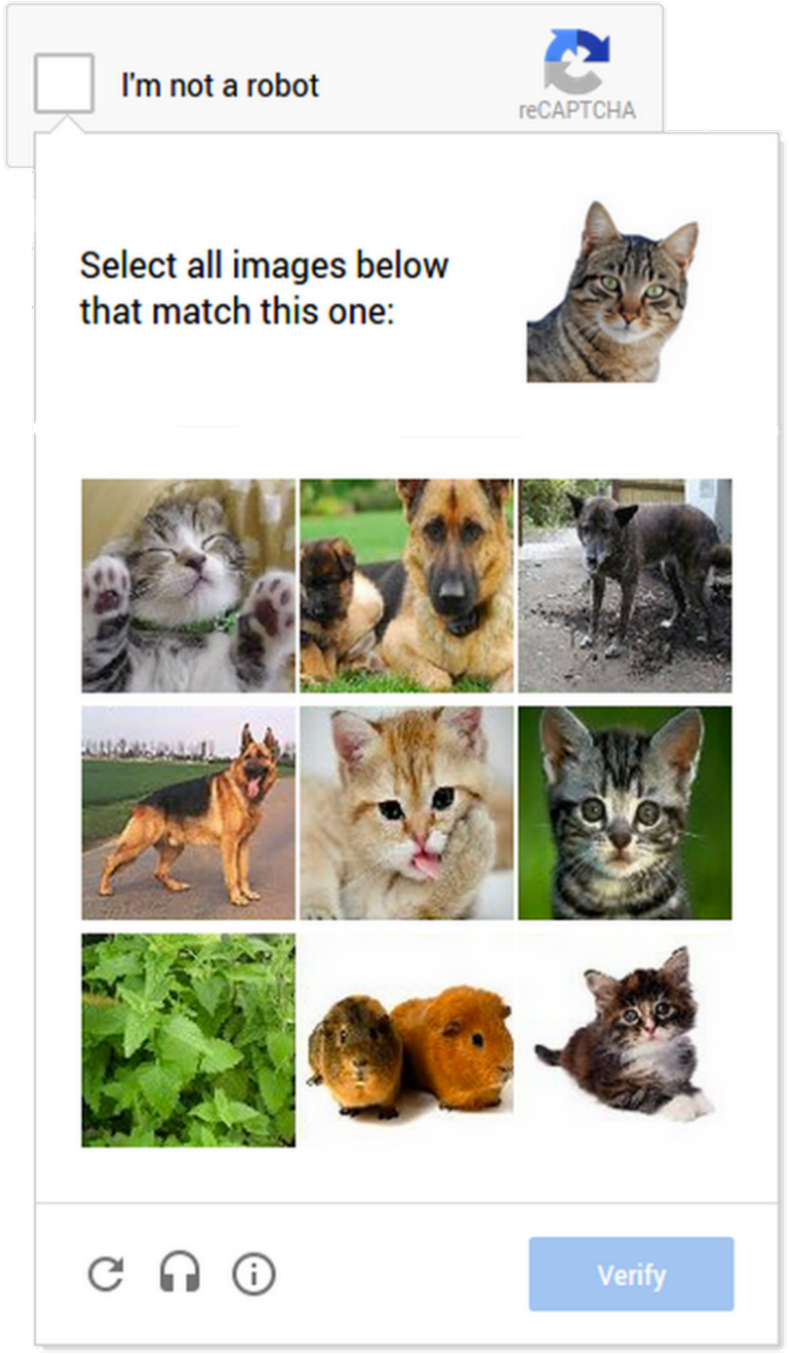
- Kassie
- Sara
- Harrison
- Trista
- Lauren
- Laura
- I'm not sure

Go to Next Photo

http://www.adweek.com/socialtimes/facebook-photos-verify/243870



http://www.wired.com/2014/12/google-one-click-recaptcha



http://www.wired.com/2014/12/google-one-click-recaptcha

Pulling humans out

Build simple rules mapping decision to response actions

Start with humans “approving” automated response

Measure rejection rate

- Decision system was wrong
- Automated response rule was wrong

Fix problems

Repeat until rejection rates are acceptably low

- (e.g. <0.1%)

Pull the human out

Circuit Breakers

Handoff to humans at particular thresholds of impact

Keep “low impact” things automated

If affects too many or critical systems: needs human review

BEDROCK
LIGHT

Distribute

Allow your users to respond to alerts

Square Sting: send alerts to users who generate them for

If user gets more than n alerts, but doesn't respond, escalate

Some alerts are not self service and go directly to infosec

90% reduction in secops load

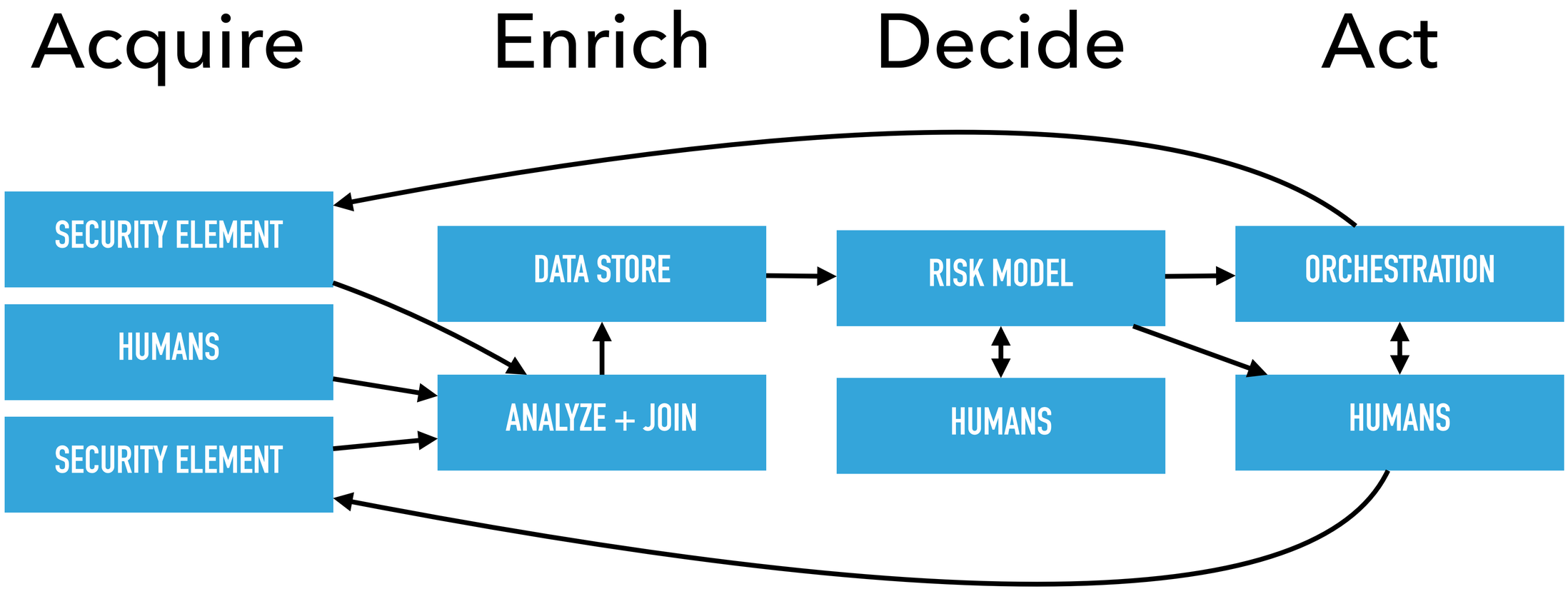
Provide context

Lookout: Shows simple description of malware (i.e. why it's bad, specific to that malware family)

- Self-remediation rate over 95% within 24 hrs

Square Sting shows the offending command line that generated the alert.

AEDA Loop


Acquire
Enrich
Decide
Act
SECURITY ELEMENT
HUMANS
SECURITY ELEMENT
DATA STORE
ANALYZE + JOIN
RISK MODEL
HUMANS
ORCHESTRATION
HUMANS

Deflate the bubble

Get Started

Set a clear initial goal

Stop compromised credentials from accessing data?

Stop malicious insiders from stealing data?

Stop compromised devices from accessing data?

Involve all stakeholders

Data architects

Developers

Sec Ops

Operations

Design

Legal

Compliance/risk

Project management

Decide your “nouns”

Subject: Device + User + App

- What do you choose to trust?

Object: Data + Applications + “Rights”

- Prefer data over applications: data sensitivity generally stays the same; application scope changes over time.

Synthesized data is hard to reason about

- aggregating data generally makes it less sensitive
- enriching data generally makes it more sensitive

Thin vertical slice

Don’t partially implement.

- Commit and don’t frag your architecture with complexity

Try to solve small problem end-to-end with endgame in mind

- Architect in big bang, rollout in small bang

Careful rollout

- If it can disrupt broader business/customers, audit first
- If only affects security team, roll out quickly

Sorry, you can’t just write a check :(

I hope this talk encourages

…vendors to build solid APIs and data models

…security engineering to rethink their security architectures

…security operations to build cyborg processes

Architecture Tips

Decouple authorization from network access

Apps should enforce explicit identity

- IP, or connectivity as proxy for real application layer authentication == pain

You don’t want to re-architect your network every time your application layer access policy changes.

Legacy apps may need wrappers if they can’t be modified.

Move trust anchors

Small trust surfaces infeasible when manually configured

When trust config emerges from data + automation,
small trust surfaces are feasible.

Decouple sensing from enforcement

Portfolios of sensors feed data system

Enforcement should be system wide,
not just on same device as sensor

Don’t trust middlemen systems

Assume they are compromised as well

- message queues, network systems

Enforce end-to-end integrity

- If you’re using a shared data bus, ensure each payload is authenticated

Validate data architectures

System-wide data flows are complex.
Do you know if they work?

- | /dev/null?
- IPS system was rigged not to receive all packets and couldn't reassemble TCP :(

A red-team with a retrospective can help you find (and fix) holes in your architecture before attackers take advantage of them.

0xFF: Apply

Next Month

- Socialize the concept of an integrated security data architecture with your team

Next 3 months

- Identify a minimum viable end-to-end closed loop you can build

Next year

- Integrate Acquisition, Enrichment, Decision, and Action processes with human approval
- If rejection rate is low, remove humans from the flow

Next 3 years

- Integrate data across your entire stack, repeatedly automating decision processes.

Thanks.