FIGHTING CYBER-THREATS WITH CROWDSOURCED INTELLIGENCE

ZOHAR DUCHIN, PRINCIPAL RESEARCHER RSA

DMBI 2016

RSA

Why Are We Loosing The Cyber Arena?

- **Security analysts as lone rangers**
  - Each analyst sees only tiny part of the picture
  - Nobody knows everything
  - Repeating mistakes that others already did

Defenders

Heroic but separated and unorganized

Attackers

Well organized and motivated (organized crime, nation state actors)

RSA

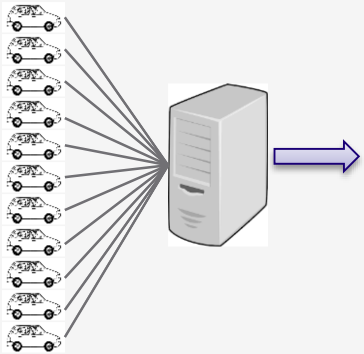
How Can Crowdsourcing Help?

Crowdsourced navigation intelligence

Tiny part of the road from each car

Analytics

Map + prediction + navigation instructions

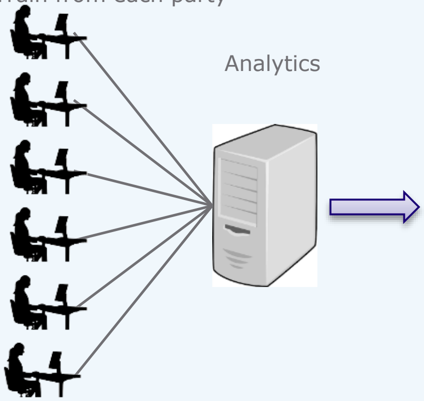


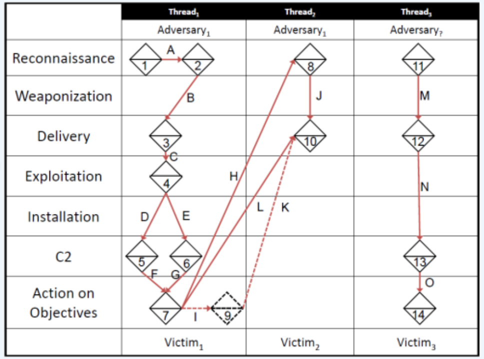
Crowdsourced security intelligence

Tiny part of security terrain from each party

Analytics

Attack threads + predictions + mitigation instructions





RSA

Helping the Analyst with Crowdsourced Intelligence

The analyst daily job

Prioritize

The ask: Help me to focus on the most important events

↓

Investigate and Decide

The ask: Guide me through the investigation process and help me to reach the right decision faster

↓

Remediate and Predict

The ask: Help me to set the right indicators for future protection
Help me to be prepared for attacks am I likely to encounter in the near future

Prioritization with Crowdsourced Intel'

- **Use cases**
  - Others opinion on the same / similar events
  - Trends of same events in the community
  - Overall reputation

- **Example: prioritizing suspicious IP addresses with community reputation**
  - Aggregating feedback from the community
  - Using lower bound interval of Wilson score to be conservative
  - Include time decay as IP addresses are dynamic

$$Score = \frac{\hat{p} + \frac{1}{2n}z^2 - z\sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}}}{1 + \frac{1}{n}z^2}$$

Where:

- n = total number of customers that have the IP *decayed with time i.e.:*

$$n = \sum_i 1\,\{customer_i\ has\ IP\} * e^{-\Delta t_{ni}/\omega}$$

- $\hat{p} = \frac{\#\ of\ risky}{n}$
- $\#\ of\ risky = \sum_i 1\{customer_i\ \text{provided}\ risky\ feedback\} * e^{-\Delta t_{fi}/\omega}$
- w = Constant that controls the speed of decay
- Δt_{fi} = age of $feedback$ from customer i
- Δt_{ni} = age of IP at customer i
- $z = 1.96$

RSA

Investigation with Crowdsourced Intel’

- **Use cases:**
  - Best practices: Investigation steps that others have taken
  - If I have found this event, what related items should I look for?
  - What is the most valuable information that will help my decision?

- **Example**
  - Filling attack kill chain using various sources in the community
    - Different customers with different detectors
  - Guiding users to investigate the missing link in the chain

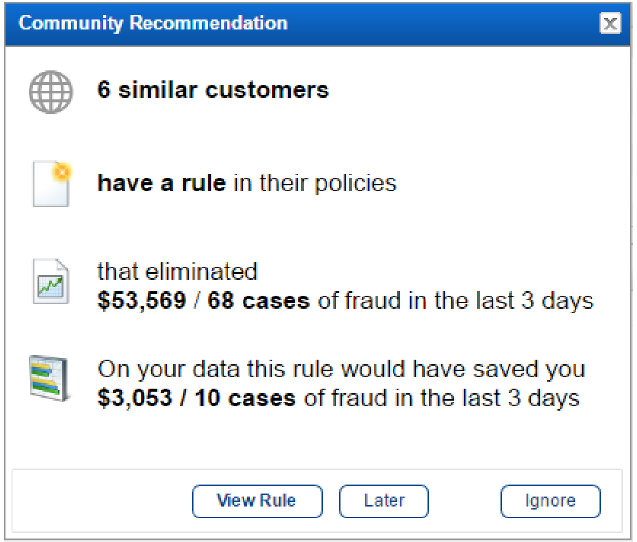
Interacting with contributors to fill missing info

Phase	Indicator	Contributed by
Reconnaissance	NA	
Weaponization	Benign File: tcnom.pdf	User C: Endpoint
Delivery	?	User B: Network
Exploitation	CVE-2009-0658 [shellcode exploiting]	External source
Installation	fssm32.exe IEUpd.exe IEXPLORE.hlp	User A: Endpoint User C: Endpoint
C2	202.abc.xyz.7 [HTTP request]	User B: Network
Actions on Objectives	Key logging	User B: SecOps

Remediate and Predict with Crowdsourced Intel'

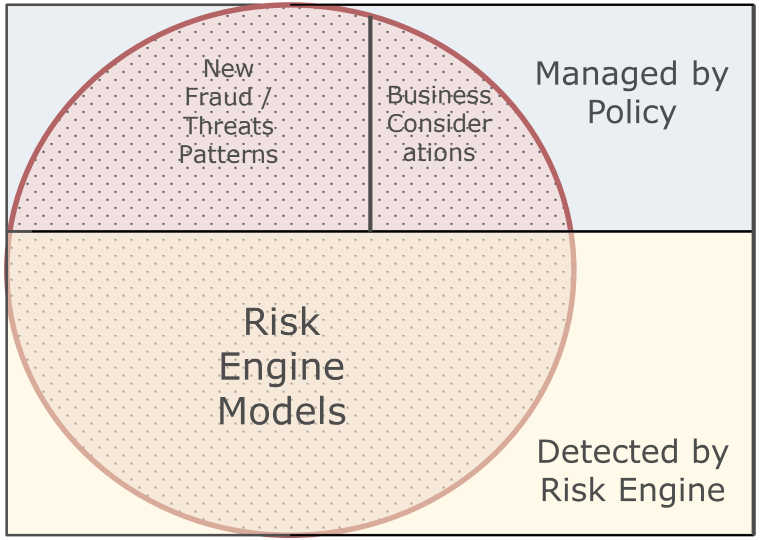
- **User cases:**
  - What customers like me have encountered
  - Recommend best known methods for protection

- **Example**
  - Recommending rules for policy of Web Fraud Detection management
  - Using user-user collaborative filtering
  - We will explore this in the next slides…



Fraud Detection Policy Overview

- **Credit card fraud detection engine**
  - Targeted to manage fraud events and business goals
  - Consists of:
    - Machine learning based risk engine
    - Policy i.e. set of rules



- What is a rule?

$$Rule = \{Conditions, Action, Meta\}$$

$$Condition = \{Sensor, Operation, Value\}$$

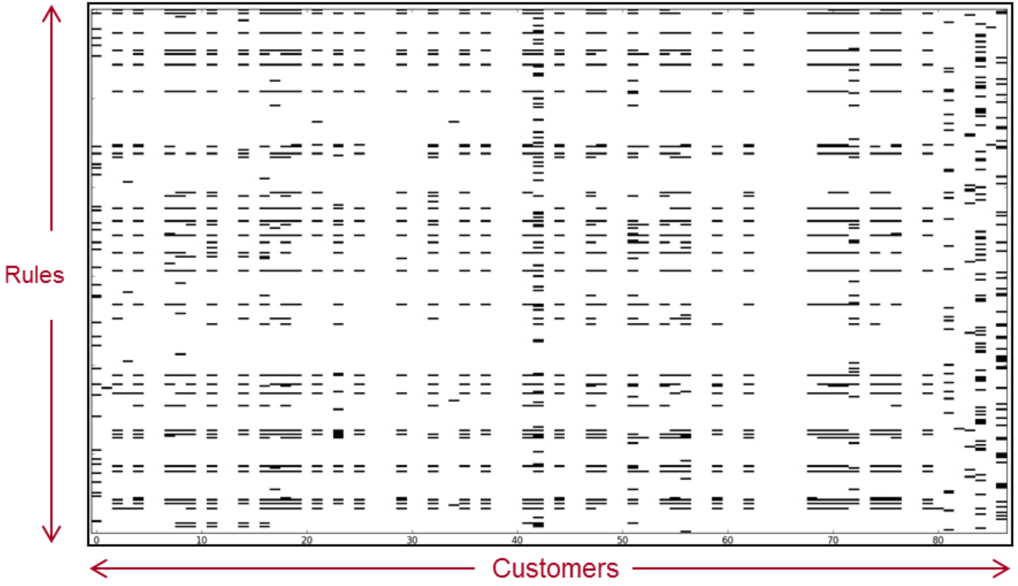
$$Action = Accept, Challenge, Block$$

$$Meta = \{Creation\ time, Fraud\ count, Fraud\ amount, False\ positive\ count\}$$

Rules Recommendation System

- **Constructing user-item (customer-rule) matrix**
  - Rules decomposition (logical or, lists)
  - Implicit rating calculation for each rule@customer
- **Similarity measure between customers**
  - Similar policy
  - Similar customers attributes
- **Find “good” rules**
  - Potentially good rating for a customer
- **Post processing**
  - Recommended rules clustering
- **Evaluation**

Visualization of customers – rules matrix



RSA

Rules Collaborative Filtering

- **“Predict” rating of a rule for specific customer based on the ratings at other customers**
  - Weighted by similarity between customers
- **Preserve each customer policy preferences**
  - Avoid mean centering with average rule performance
- **Measure similarity between customers according to:**
  - How similar are their policies
  - How similar is their context
- **Selecting the rules with the highest rating**
  - Also passing a threshold that is specific to each customer

Predicted rating of rule i at user a

Rating of rule i at user u

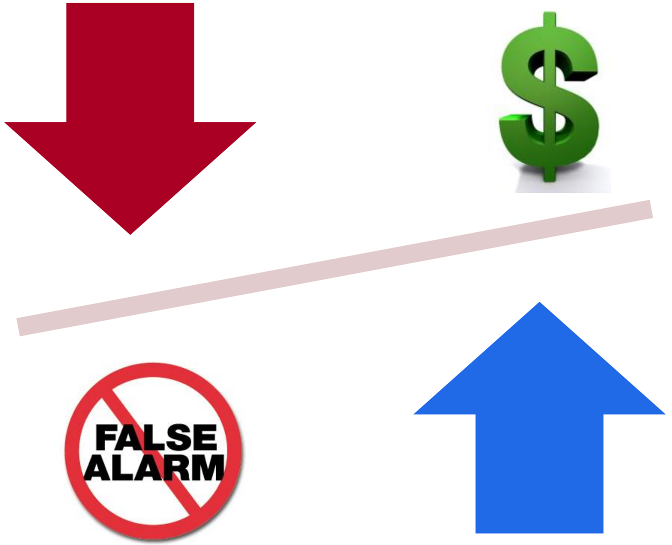
Similarity between user a and user u

$$P_{a,i} = \frac{\sum_{u=1}^{n} r_{u,i} * w_{a,u}}{\sum_{u=1}^{n} w_{a,u}}$$

$$w_{a,u} = \underbrace{\frac{0.5(\vec{R}_a * \vec{R}_u)}{\|\vec{R}_a\| * \|\vec{R}_u\|}}_{\text{Similarity between rule sets}} + \underbrace{\frac{0.5(\vec{M}_a * \vec{M}_u)}{\|\vec{M}_a\| * \|\vec{M}_u\|}}_{\text{Similarity between customers' meta (location, industry, size, …)}}$$

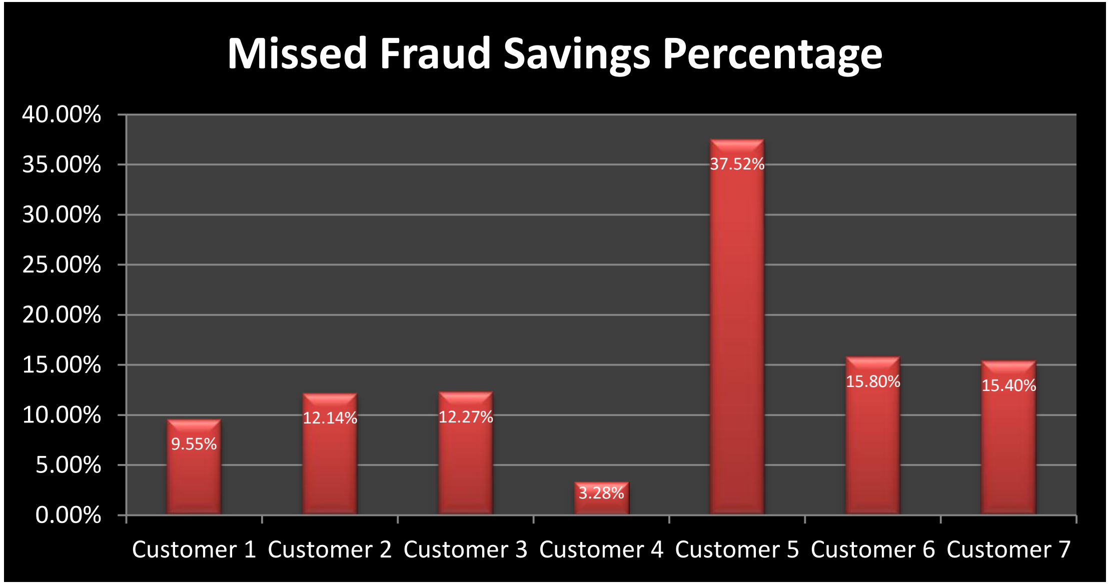
Evaluation

- **Metrics should be specific to application**
  - In this case fraud detection
- **Key performance indicators are:**
  - <u>Amount</u> of money savings due to missed fraud detection – the higher the better
  - <u>Count</u> of false alerts – the lower the better
- **Each customer has its own preferences**
  - $1000 may high amount for one customer and low amount for another
  - 10 false alerts may be too high for one customer and acceptable for another
- **In the end of the day, online evaluation protocol is needed to fine tune the model**



Rules Recommendation POC Results

- **Data**
  - 87 customers
  - 306 rules
  - 4 months transactions



- **Average increase in fraud savings: 15% (and up to 37%)**
  - Adding only 9 false alerts over the test period

Summary

- **Intelligence sharing is a key for fighting cyber attacks effectively**

- **Current intelligence sharing is very basic and manual; it is time for crowdsourcing and advanced analytics to step in**

- **Crowdsourcing can be leveraged in all levels of the security analyst work**
  - Prioritization
  - Investigation
  - Prediction / remediation

- **All these are enablers for high level co-operation that can keep the good guys one step ahead of the bad guys**

RSA

Thank You

Email me: zohar.duchin@rsa.com