UMD Lablet Overview

Jonathan Katz
Computer Science

Michel Cukier
Reliability Engineering

Lablet overview

- 20 faculty researchers
  - 15 from UMD, 5 external collaborators
- UMD faculty drawn from five different departments on campus
  - CS, ECE, Information Studies, Criminology, Reliability Engineering
  - Collaboration fostered by the Maryland Cybersecurity Center (MC2)

Lablet members

- Adam Aviv (USNA)
- John Baras
- Marshini Chetty
- **Michel Cukier (Co-PI)**
- Tudor Dumitras
- Jeff Foster
- Jen Golbeck
- Michael Hicks
- David Van Horn
- Joseph JaJa
- **Jonathan Katz (PI)**
- Dave Levin
- David Maimon
- Michelle Mazurek
- Babis Papamanthou
- Aditya Prakash (VA Tech)
- VS Subrahmanian
- Mohit Tiwari (UT Austin)
- Sam Tobin-Hochstadt (IU)
- Poorvi Vora (GWU)

Hard problems

	Scalability	Policy	Metrics	Resilient	Human
Does the presence of honest…					X
Reasoning about protocols …					X
Understanding developers' …					X
Understanding how users …					X
Empirical models for …			X		
Human behavior and cyber …			X		X
Measuring and improving …			X		X
User-centered design for security			X		X
Trustworthy and composable …	X				
Trust, recommendation systems …	X	X			X

Understanding how users process security advice

Task lead: Michelle Mazurek

Hard problem(s): Human Behavior

Summary of activities

- Where do users learn advice?
  - Interview study; confirmatory survey (S&P, CCS 2016)
- How do socioeconomics affect security behavior?
  - Random-digit-dial survey data (CHI 2017)
- How can we improve behaviors?
  - "Edutainment" video to promote updating
    - Poster at NDSS 2017; paper to be submitted CHI 2018
  - Comparison of 2FA enrollment messages
    - SOUPS 2017 workshop; full paper to be submitted CHI 2018

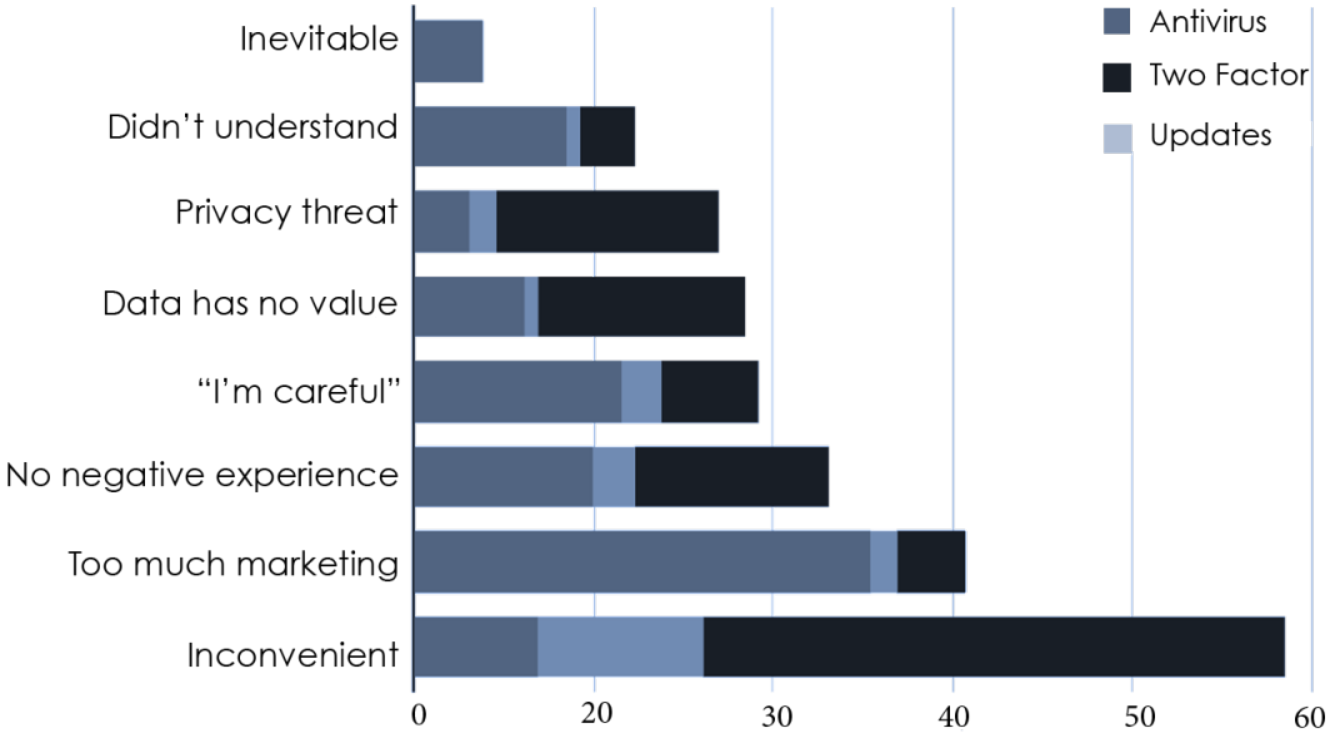
Learning secure behavior

- Advice is stratified by educational attainment
- Trust of source is often key
- Advice rejected for convenience, but also marketing, privacy
- Negative experiences as learning tools
- For 2FA, messaging is less important than perceived account value

Fiction as learning experience

“I put a password on my WiFi network after watching a TV show. It showed people going by houses and WiFi snooping . . . shows like that, they make you think.”

Why reject advice?



Convenience matters, but it's not the only thing

User-centered design for security

Task leads: Jen Golbeck and Adam Aviv

Hard problem(s): Human behavior, security metrics

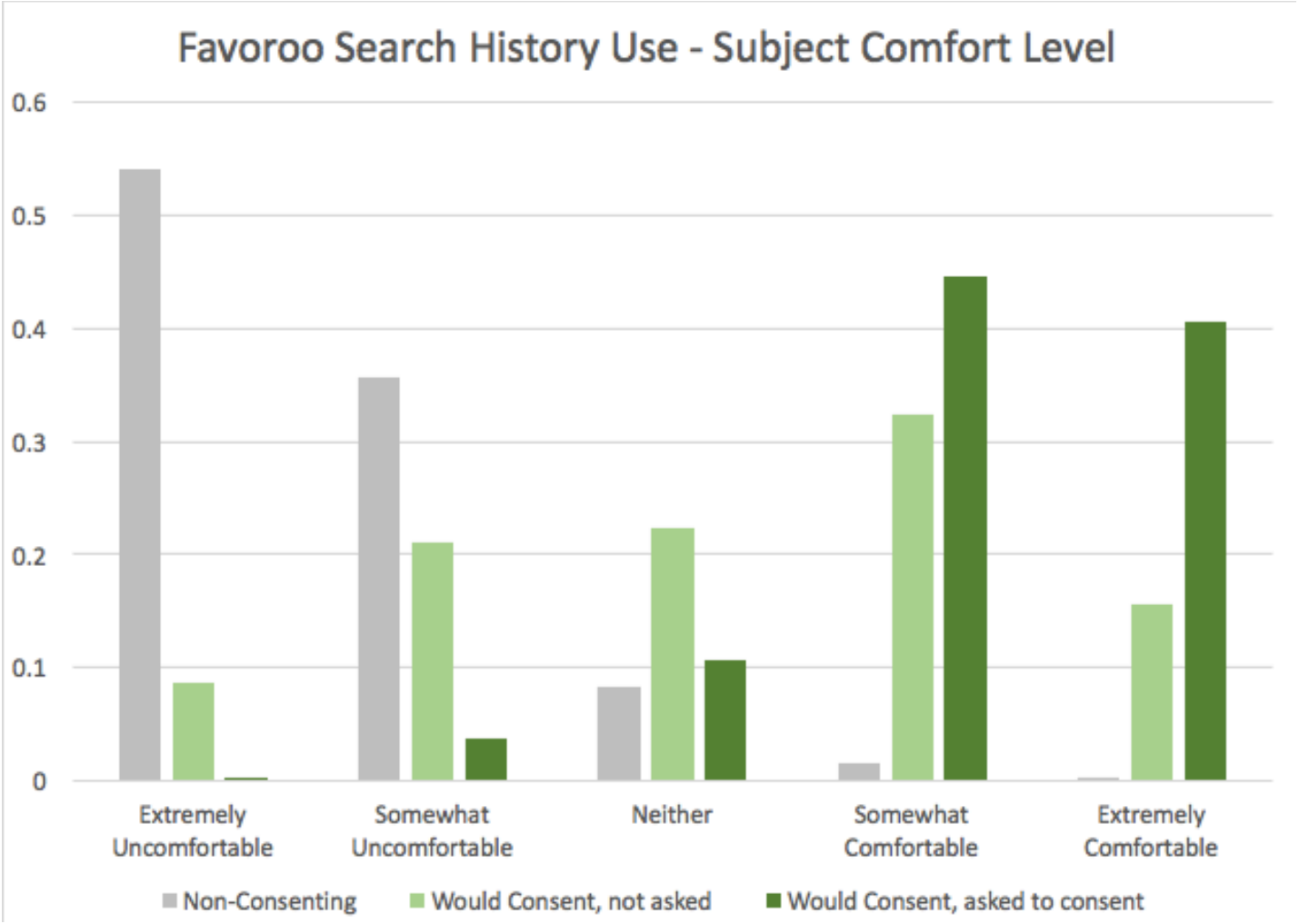
Understanding User Preference & Consent

- Two studies
  - Recommender system data
  - Public wifi hotspots
- Conclusions from both
  - People want transparency in how their data/resources are used
  - People want the option to consent
  - When they cannot consent, they are less comfortable with the use and feel they are at greater risk

Recommender Systems and Personalization

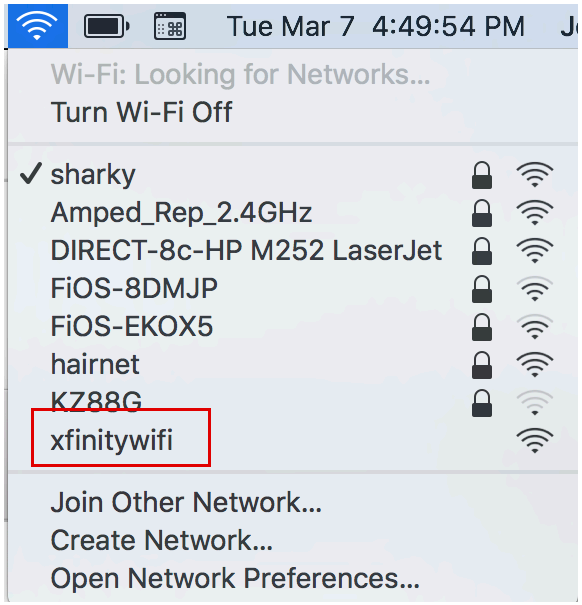
- Use people's data to recommend and personalize content
- How comfortable are people with different data being used this way?
  - Would they consent to it?
  - How would they feel if it were used without first giving consent?
- We created a fictional app, Favoroo, and asked people about different data points
  - 662 subjects on mturk

Main insight: Even if people would consent, they want to be asked

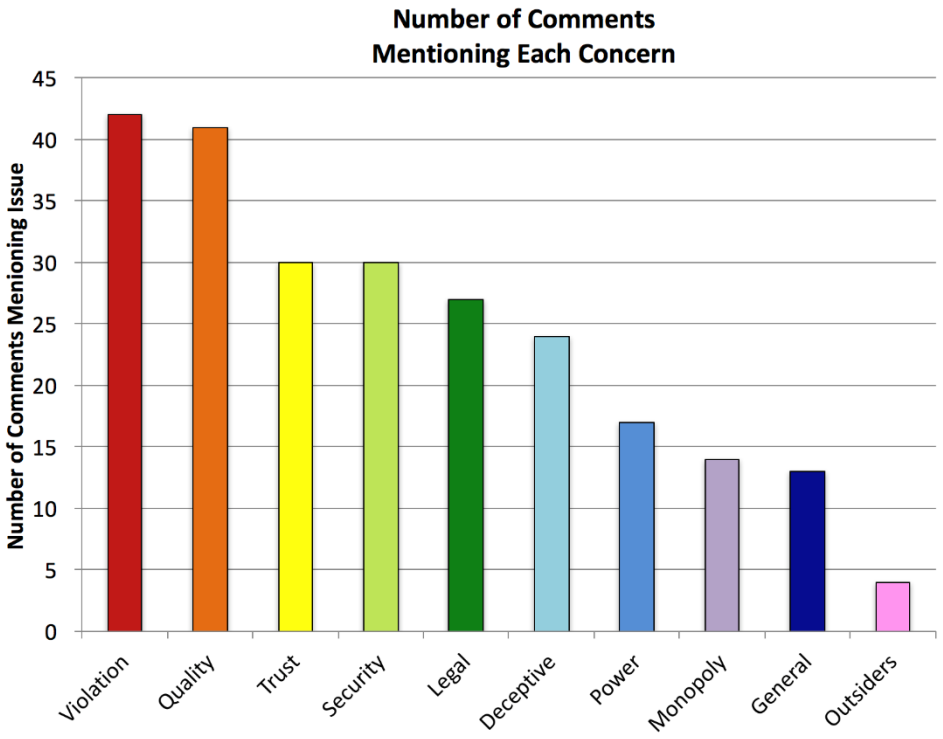


Public WiFi Hotspots

- ISPs are installing public wifi hotspots in the personal in-home routers of millions of customers
- As a rule, they do not tell customers they are doing this and, in some cases, they do not let customers opt out
- We reviewed 501 online comments on news articles about the practice to discover user concerns



89% of commenters were concerned



- Violation of Privacy
- Quality of service impacts
- Do not trust ISP to be responsible
- Security risks
- Legal risks for outsiders' bad behavior on my router
- Practice is deceptive
- Power costs increase with hotspot running
- Monopolistic practices by ISP
- General, non-specific worries
- Outsiders may come creeping around

Does the presence of honest users affect intruders' behavior?

Task leads: Michel Cukier, David Maimon

Hard problem(s): Human behavior

Social sciences and cybersecurity

- **Idea:** Investigate application of criminological theories to cybersecurity
  - Routine activity theory
  - Rational choice theory
  - Deterrence theory

Research questions

- What is the effect of a warning banner on system trespassers' online activities?
- What is the effect of a surveillance banner and/or process on system trespassers' online activities?
- What is the effect of legitimate users on system trespassers' online activities?

Deterrence summary

Warning

- Reduces significantly the duration of the sessions

Surveillance

- Impact whether commands are typed in first session

User

- Impact number of trespassing events when an admin user is present

Reasoning about protocols with human participants and physical objects

Task leads: Jonathan Katz, Poorvi LVora

Hard problem(s): Human behavior, resilient architectures

Internet Voting: Apollo

Dawid Gawel, Maciej Kosarzecki, Poorvi L. Vora (GW), Hua Wu (GW), Filip Zagórski

- Existing internet voting schemes are either:
  - Vulnerable to credential-stealing attacks

    Helios, used by ACM and IACR for annual elections
  - Difficult to use

    Remotegrity, used in Takoma Park city election, 2011
- We propose a hybrid, Apollo, with the strengths of both
- Both are end-to-end-verifiable (E2E-V), as is Apollo

Credential Stealing

- Voter votes remotely
- Voting terminal participates honestly
  - UNTIL it gets the credential to cast the vote
  - It then replaces the vote
- End-to-end-verifiable (E2E-V) protocol enables the careful voter to detect the problem:
  - but she cannot prove it
- If the voter complains, is there a problem?

Our Contributions

- Implementation vulnerabilities in Helios
  - Motivated by our findings, code is now patched
- Apollo, extension of Helios
  - Prevents credential stealing
  - Votes not authorized by the voter cannot be included in tally
  - Voter can prove her vote was changed
  - Simple subprotocol for voter interaction with devices used to audit the voting system

Empirical models for vulnerability exploits

Task lead: Tudor Dumitras

Hard problem(s): Security metrics

Research

- Derive empirical models of vulnerabilities and attack surfaces; correlate with real-world attack data
  - What vulnerabilities are exploited in real world?
- Understand deployment-specific factors that influence security of real systems
  - How to best characterize *attack surface*

- *Using real-world field data from WINE*

Measuring security of deployed systems

- Count of vulnerabilities exploited
- *Exploitation ratio*: ratio of exploited vulnerabilities to disclosed vulnerabilities
- *Survival probability*: time to exploit
- *Exercised attack surface*: number of distinct exploits on a host per month

Exploitation ratio [RAID'14]

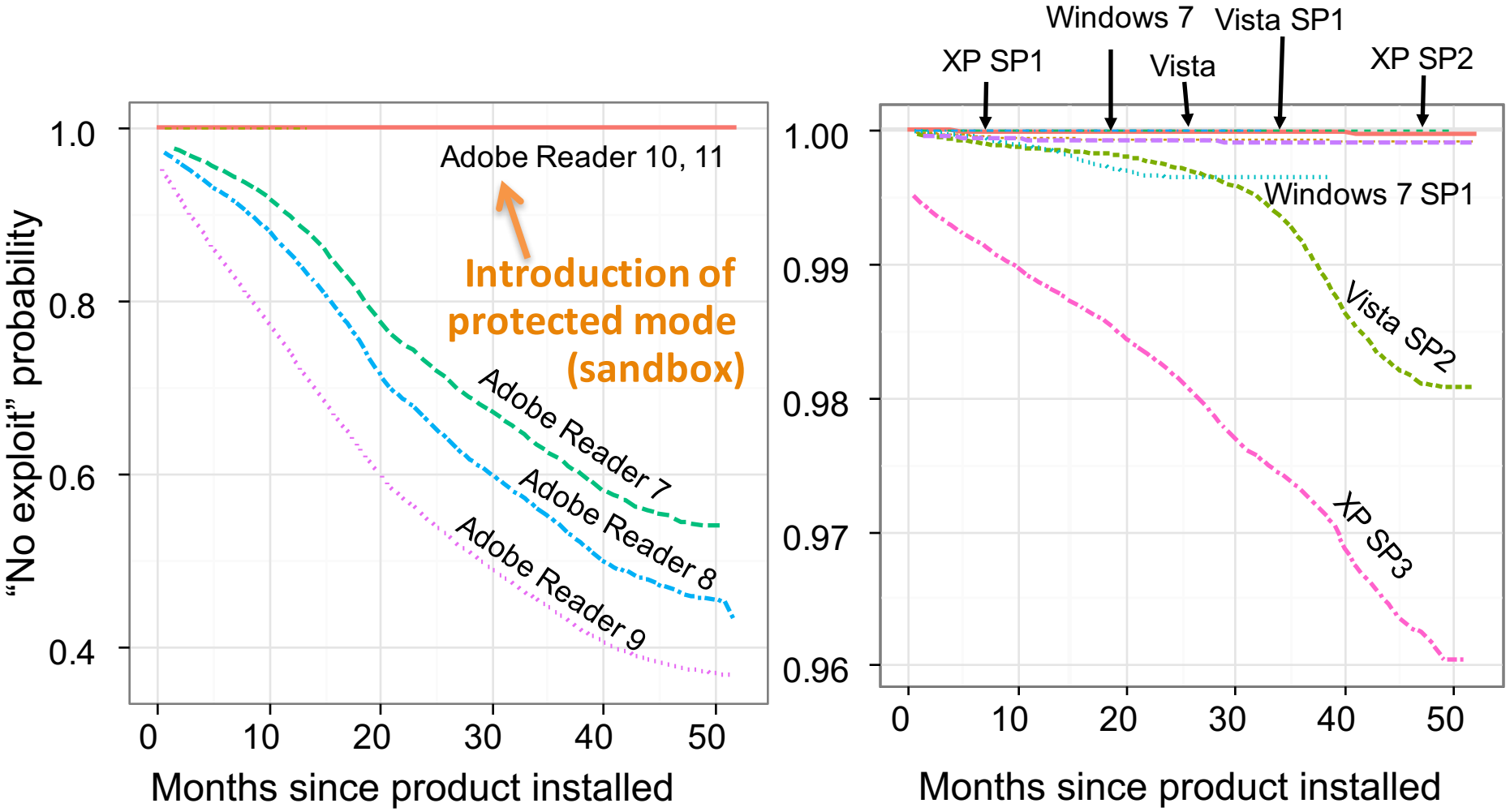
- Identify exploits from Symantec signature definitions
  - http://www.symantec.com/security_response/threatexplorer/azlisting.jsp

Product	Exploited Vulnerabilities	Exploitation Ratio
Office 2000	26	0.32
Office 2003	41	0.36
Office 2007	17	0.31
Office 2010	4	0.29
Adobe Reader 6	5	0.21
Adobe Reader 7	11	0.17
Adobe Reader 8	29	0.16
Adobe Reader 9	29	0.11
Adobe Reader 10	12	0.09
Adobe Reader 11	4	0.07

Fewer than 40% of known vulnerabilities are exploited

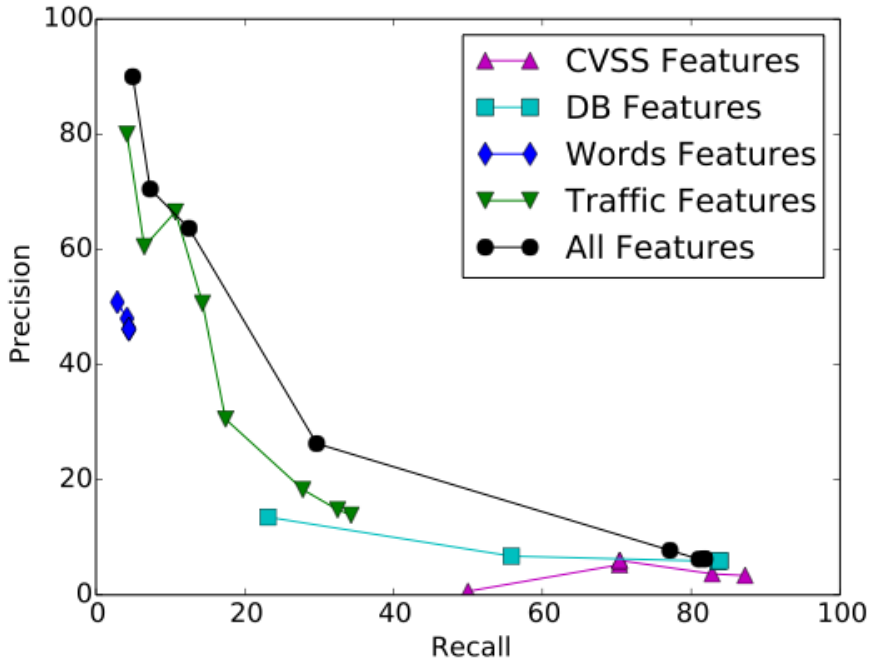
Decrease with newer versions

Time-to-exploit [RAID'14]



Predicting Exploits in the Wild [USENIX Security'15]

- Trained classifier with multiple features
  - Vulnerability specific (e.g. CVSS score, vulnerability type)
    - High recall, low precision
  - Influence specific (e.g. terms used on Twitter, retweet traffic)
    - High precision, low recall



Implications

- Scarcity of exploits matches cybercrime data
  - 2013: $100,000 per zero-day exploit
- Reasons?
  - System-security technologies that render exploits less likely to work
  - Commoditization of malware industry
- Take-aways?
  - Prioritization of patch deployment
  - Risk assessment

Human behavior and cyber vulnerabilities

Task leads: VS Subrahmanian, Tudor Dumitras, Marshini Chetty

Hard problem(s): Security metrics, human behavior

Research

- Characterize the rate of vulnerability patching
- Determine the factors that influence the rate of patch deployment
  - Technological
    - Using WINE dataset
  - Sociological
    - Based on targeted user studies http://netchi.umd.edu/software-updating-study.html

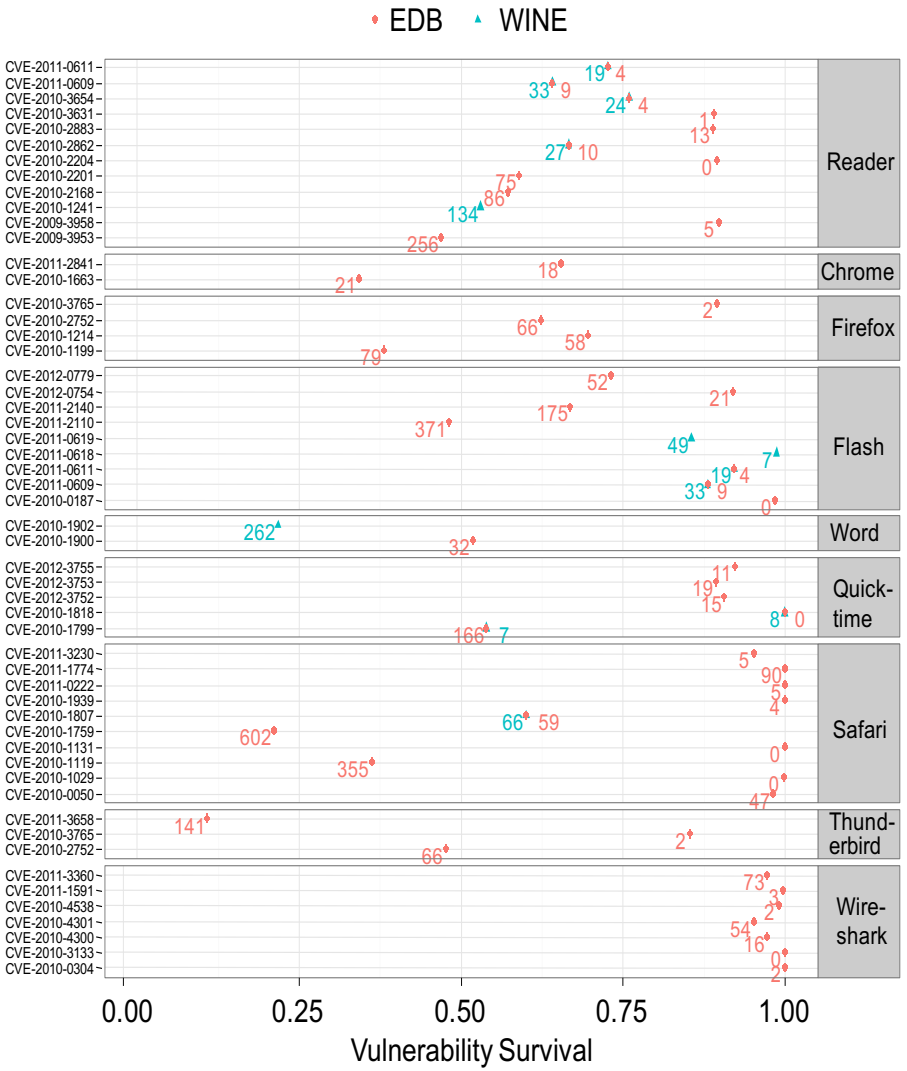
Measuring Vulnerability Patching [Oakland'15]

- Collected largest corpus of vulnerability patch measurements
  - **1,593** vulnerabilities in **10** client applications on Windows
  - Client-side applications: difficult to measure using known techniques (e.g. network scanning)
    - Chrome, Firefox, Opera, Safari (browsers)
    - Flash Player, Quicktime (multimedia)
    - Thunderbird (email)
    - Adobe Reader (document reader)
    - Microsoft Word (editor)
    - Wireshark (networking)
  - Often targeted in spear-phishing

- Daily measurements of vulnerable host population
  - Observation period: **January 2008 – December 2012**
  - Diverse patching patterns observed

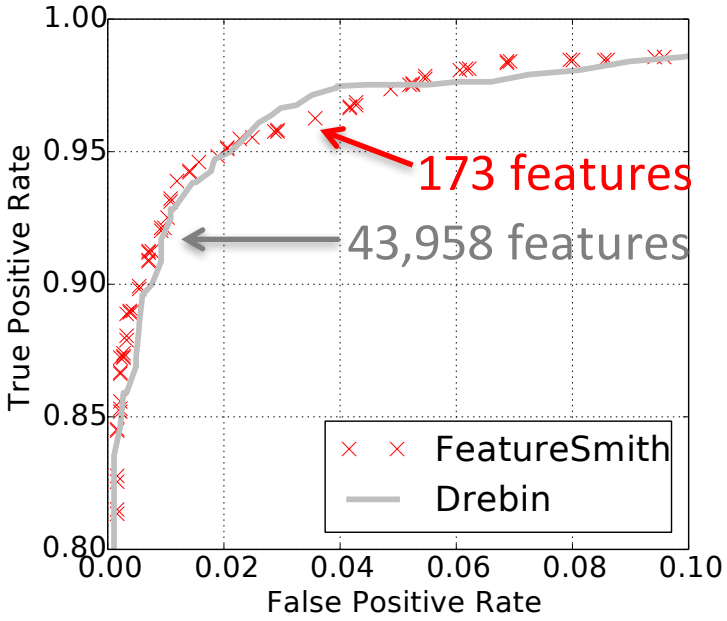
Patch Levels At Exploit Time [Oakland'15]

- Median percentage of hosts patched: **14%**
  - Considering both proof-of-concept and real-world exploits
  - Only one real-world exploit found more than 50% hosts patched
- These numbers must be interpreted as **upper bounds**



Automatic Feature Engineering [CCS'16]

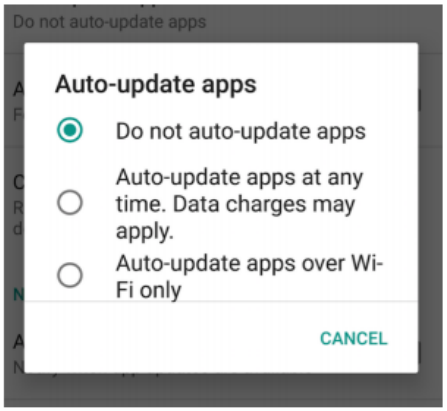
- Learn the threat semantics by mining the security literature
  - Integrate human mental models in machine-learning based detectors
- Detection performance **on par** with state-of-the-art detector (Drebin) [Arp+, NDSS'14]
  - Uses manually engineered features
- Discovered **new features**, missing from the manually engineered set
  - API calls that leak private information
  - Allow us to detect `Gappusin` family (FN for Drebin)



Implications

- Patch deployment exhibits a **long tail**
  - Automated updates faster
- Exploits are effective **even if not zero-day**
- Measurement corpus provides empirical data for **future modeling efforts**
  - Risk management, cyber insurance
- Can discover threat semantics automatically
  - Ensure that ML models are based on meaningful features rather than data artifacts

Users + Software Updates



- **Goal:** To understand how user characteristics are associated with attitudes towards auto-updates on mobile phones
- **What we did:** Surveyed 477 Android users on Amazon Mechanical Turk in Spring 2016

Mathur, A. and Chetty, M. *Impact of User Characteristics on Attitudes Towards Automatic Mobile Updates*. To appear at SOUPS 2017 in Santa Clara, CA, USA

What we found:

1. Users who avoid mobile auto-updates have had a negative experiences with updates before, but these negative experiences may not have been on the mobile device
2. Users who avoid auto-updates also take fewer financial investment and ethical risks

What we found (cont.):

3. Users who avoid auto-updates also exhibit a greater propensity for being proactive about their online security
4. Users more comfortable auto-updating security updates on Android if they perceived an app as trustworthy

What do we recommend:

1. Allow mobile users to roll back app updates to encourage them to be more risk taking with applying updates
2. Leverage user characteristics to personalize nudges + messages to encourage auto-updating on mobiles
3. Study others in software updating ecosystem to minimize negative update experiences for users on all platforms

Trustworthy and composable software systems with contracts

Task lead: David Van Horn

Hard problem(s): Composability

Trustworthy and Composable Software Systems with Contracts

- Language-based security mechanisms make two unrealistic assumptions:
  1. analyzed code comprises a complete program (as opposed to a framework or set of components)
  2. software is written in a single programming language

Trustworthy and Composable Software Systems with Contracts

- These assumptions ignore the reality of modern software:
  - composed of large sets of interacting components
  - constructed in several programming languages that provide varying degrees of assurance that the components are well-behaved

Trustworthy and Composable Software Systems with Contracts

- Project addresses limitations by developing new static-analysis techniques based on **software contracts**, providing a way to extend analysis of components to reason about security of entire heterogeneous system.
- Hard problems:
  - Scalability and Composability
  - Security-Metrics-Driven Evaluation, Design, Development, and Deployment

Trustworthy and Composable Software Systems with Contracts

- Results:
  - Theoretical framework for contract verification
  - Theory is proven sound and relatively complete
  - Robust implementation of contract verifier
  - Empirical evaluation shows effectiveness
  - Applied to approach to multi-language programs
- Papers: ICFP'14, PLDI'15, JFP'17, in submission to POPL'18

Trustworthy and Composable Software Systems with Contracts

- Community interaction, education:
  - Presented at NII, Dagsuhl, & top-tier conferences
  - Tutorial at POPL
  - Lecture series at Oregon grad PL summer school
  - Included in UMD grad class on Program Analysis
  - Tutorial at PL Mentoring Workshop
  - To be included in intro prog class starting Fall 2017

Trust, recommendation systems, and collaboration

Task lead: John Baras

Hard problem(s): Policy-governed collaboration, human behavior

Using Trust in Distributed Consensus with Adversaries in Sensor and Other Networks

- **In distributed systems or networked systems**: consensus is reaching agreement regarding the states of agents using only local information – used to compute collaboratively some functions.
- Analysis of **distributed consensus with Byzantine adversaries** is becoming increasingly important in both distributed computing and networked (control and communication) systems.
- Sensor networks and fusion is a good application/example

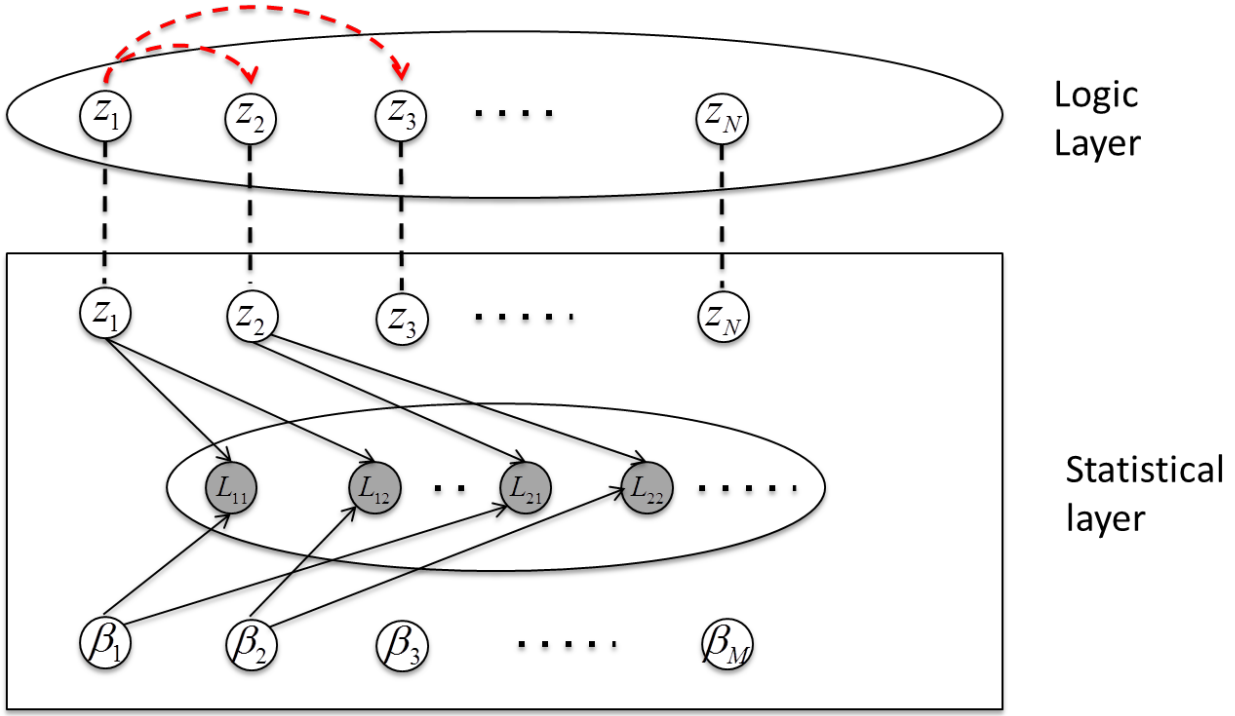
Results

- Developed trust model with various decision rules based on local evidence in the setting of Byzantine adversaries.
- Trust-Aware consensus algorithm proposed is flexible; it can be extended to more complicated trust models and decision rules.
- Simulations show our algorithm can effectively detect malicious strategies even in sparse networks of *connectivity* < $2f+1$; f is the number of adversaries.

Incorporate Domain Knowledge into Trust Aware Crowdsourcing

Developed and applied generalized probabilistic soft logic framework (GPSL) that contains two kinds of rules:

- First-order logic rules
- Special cost-function rules, the bridge between logical layer and statistical layer



Experimental Results

- Benchmarks
  - TCDK: trust-aware crowdsourcing with domain knowledge
  - TC: trust-aware crowdsourcing (without domain knowledge)
  - MV: majority voting
- Affective Text Analysis Dataset:

$$\lambda : 5.0,\ \ \mathrm{tl}(Q,X) \wedge \mathrm{oppRel}(X,Y) \Rightarrow \neg\, \mathrm{tl}(Q,Y)$$
$$\lambda : 1.0,\ \ \mathrm{tl}(Q,X) \wedge \mathrm{simRel}(X,Y) \Rightarrow \mathrm{tl}(Q,Y)$$
$$\lambda : 1.0,\ \ \mathrm{LinearLoss}(\beta, \mathrm{tl}(Q,X))$$

Model	Precision	Recall	F1	Accuracy
TCDK	31.91	75.00	44.48	93.83
TC	34.04	51.61	41.03	92.33
MV	34.04	47.06	39.51	91.83

STAR: Semiring Trust Inference for Trust-Aware Social Recommenders

Challenges

Sparse connections in trust network

Trust data availability

- Reluctance in disclosing trust information

Inconsistency and conflicts in trust opinions

- Trust opinions are 'local'
- Trust and distrust

Nonlinearity in trust formation

Key Idea

Transform intuitive heuristics into formal modeling

- Trust propagation
- Trust aggregation

Nontrivial to approach

- Distrust
- Nonlinearity
- Conflicts

An algebraic structure $\langle A, \oplus, \otimes, \mathbf{0}, \mathbf{1} \rangle$

- Additive ($\oplus$) and multiplicative operation ($\otimes$)
- E.g. nonnegative integer set with normal addition and multiplication
- Well developed tool in constraint satisfaction problems (CSPs)

Semiring model for trust inference

- Addition $\longleftrightarrow$ aggregation
- Multiplication $\longleftrightarrow$ propagation

Comparison with Others -- Results

Method	Error rate
STAR	5.8%
Graph-theoretic linear approach (Guha et al.)	6.4%
Machine learning approach (Leskovec et al.)	~6%
Probabilistic confidence model (DuBois et al.)	6%

Epinions trust network dataset for experiments

- Largest dataset available

Improvement on accuracy obtained using the STAR approach, with computation efficiency and interpretability

Developed and analyzed STAR

- Algebraic approach for trust inference based on semiring structure
- Better interpretability, efficiency and performance
- Trust iteration + partial reciprocity for performance improvement

Experiments on real-world dataset

- With advantages in accuracy and coverage

R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In WWW, 2004

J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In WWW, 2010

T. DuBois, J. Golbeck, and A. Srinivasan. Predicting trust and distrust in social networks. In PASSAT and IEEE SocialCom, 2011

Questions?