

Selected Topics for IT Security

Applied Machine Learning for Cyber Security An Overview

Lecturer
Huang Xiao

xiaohu (at) in (dot) tum (dot) de

Chair for IT Security (I20), Prof. Dr. Claudia Eckert
Technische Universität München

Course Information

- Vorlesung: 2 SWS / 4,0 ECTS-Credits
- Veranstalter: Claudia Eckert
- Zeit und Ort: Mi, 17:00 – 19:00 Uhr, 00.013.009
- Beginn: Mittwoch, 13.04.2016
- Ende: Mittwoch, 13.07.2016
- Prüfung: Mündliche Prüfung
- Website: <https://www.sec.in.tum.de>

Questions to the course, please contact:

Thomas Kittel (kittel@sec.in.tum.de)

Topics

- 13.04.2016: Huang Xiao – Applied Machine Learning on Cyber Security: An Overview
- 20.04.2016: Huang Xiao – Adversarial Learning: AI as a New Security Target
- 27.04.2016: Thomas Kittel, Sergej Proskurin – Virtual Machine Introspection
- 04.05.2016: Alexander Malkis – Proving security properties of multithreaded programs
- 11.05.2016: Steffen Wagner – Trusted Computing: TPM2.0
- 18.05.2016: Julian Kirsch – Semi-automated Reverse Engineering Techniques Targeting Contemporary Malware
- 25.05.2016: Julian Schütte – Android Application Security
- 01.06.2016: George Webster – Large Scale Malware Analysis
- 08.06.2016: Apostolis Zarras – Internet of Things: A Playground for Hacking
- 15.06.2016: Johann Heyszl – Embedded System Security: Common Pitfalls and Solutions
- 22.06.2016: Andreas Ibing – Symbolic Execution for Automated Bug Detection
- 29.06.2016: Sascha Wessel, Michael Weiß – Container-Technologie, OS Architectures, trust-me solution
- 06.07.2016: Paul Muntean – Control Flow based Security Techniques
- 13.07.2016: TbA

Outline

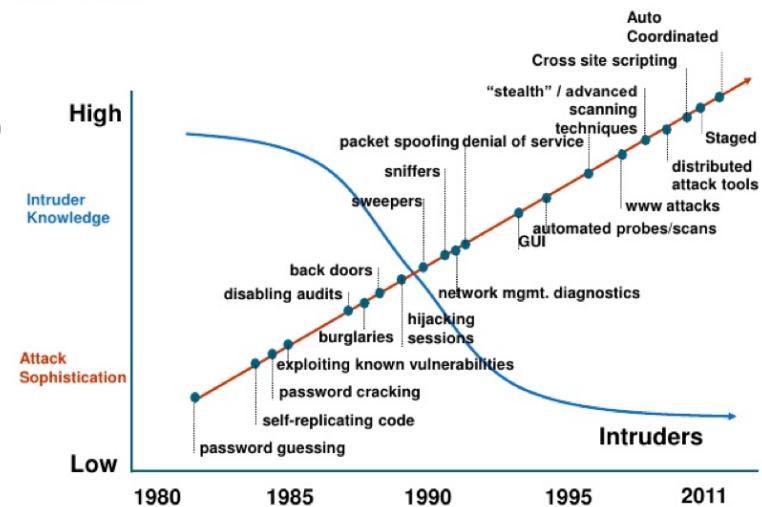
- **Introduction & Motivation**
- Machine Learning for Security
- Research Subfields
- Challenges & Discussion
- References

Introduction

- Information Technology as central role
- Process, interpretation, transmission of information
- Built on technological stacks, e.g., networks, sensors, computers
- No perfect system exists
- Information security is “good enough” solution
- Goals: C-I-A triad
- Data-driven Security

Motivation

- Attack sophistication is increasing
- Required expertise is decreasing
- Quality of tools is improved constantly
- Technological trends:
 - Big data, Cloud, IoT, Industry 4.0



Why now

- Security Analysts are often overwhelmed
- Data collection & Storage & Retrieving is well underway
- Current detection techniques might fail on
 - Polymorphic malwares
 - Zero-day attacks
 - APTs
- Dissolving network perimeter:
 - BYOD/Cloud

Generations of Security

Intrusion Detection Systems (IDSs)

- 
- Layered security
 - High false positive

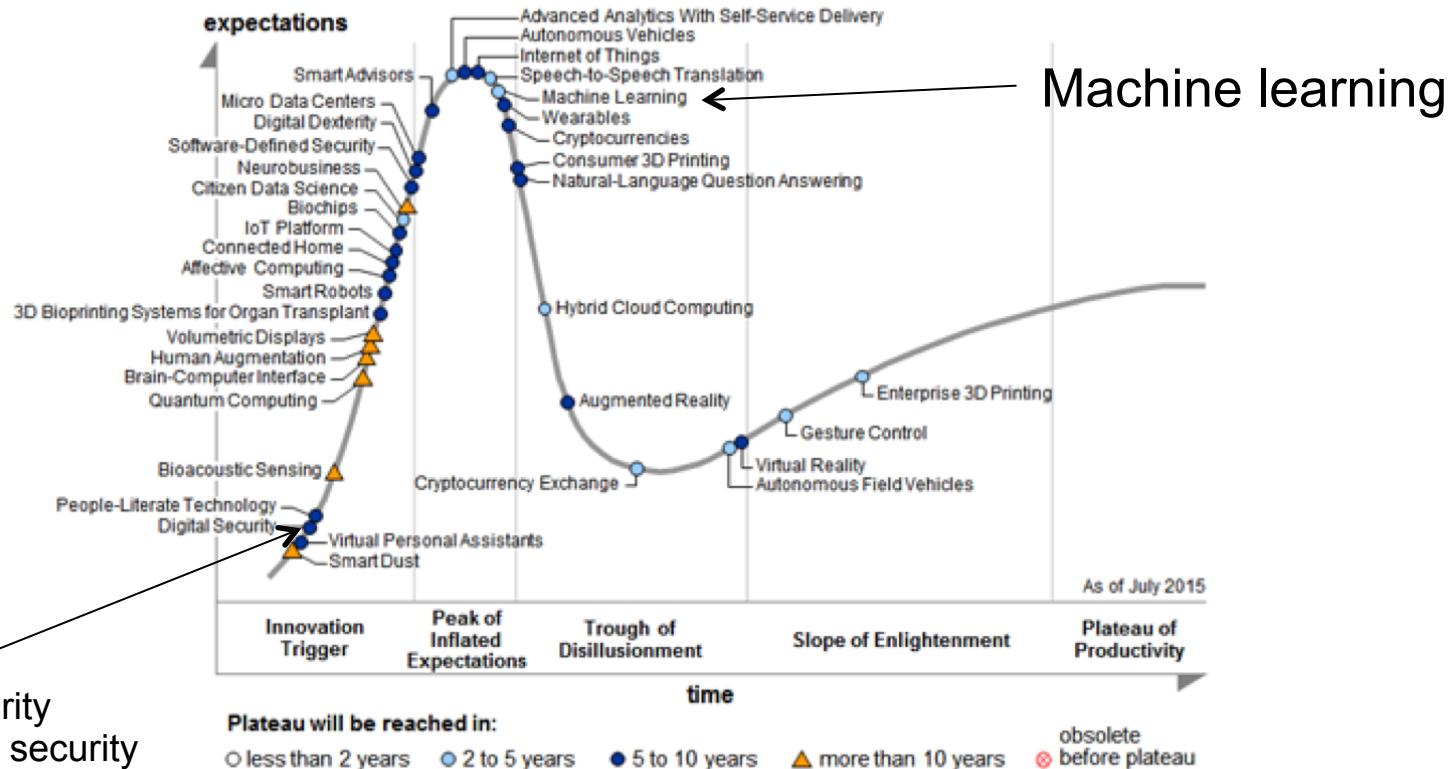
Security Information and Event Management

- 
- Alert correlation
 - Actionable Information

Big data analytics for security

-
- Context awareness

Technology hype ~5 years



- ✓ Digital security
- ✓ Data driven security

Source: Gartner (August 2015) Hype cycle

Digital Security

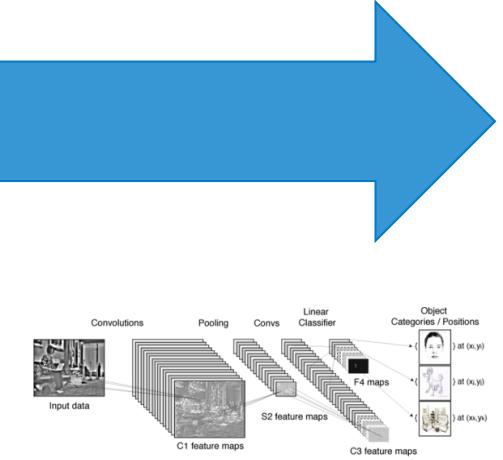
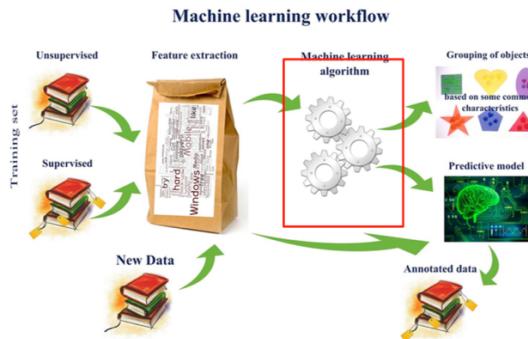
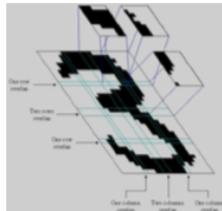
- Rising research direction in IT security
- Driven by Machine learning / data mining techniques
- Huge amount of data generated
- 2010 Verizon: **86%** cases of breach, evidence was in logs
- How do we make sense of data?
- What machine learning can not do for now? **Cyber Security**

Outline

- Introduction & Motivation
- **Machine Learning for Security**
- Research Subfields
- Challenges & Discussion
- References

Introduction of Machine Learning

Fast pacing AI development



~70s-80s: Pattern Recognition

- ✓ Based on rules, logic, expert knowledge
- ✓ Automate tasks by program

~90s – now: Machine learning

- ✓ Uncertainty introduced by Pearl et al.
- ✓ Learning models from seen data samples (historical view)
- ✓ Focus on prediction
- ✓ E.g., SVM, decision tree, Naïve Bayes, etc...

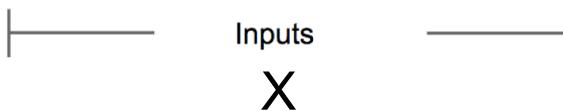
2006 - now: Deep learning

- ✓ Renaissance of NNs
- ✓ Unsupervised feature learning
- ✓ Learning non-local patterns
- ✓ Mimic how cortex works
- ✓ No theoretical ground so far
- ✓ Works like a charm

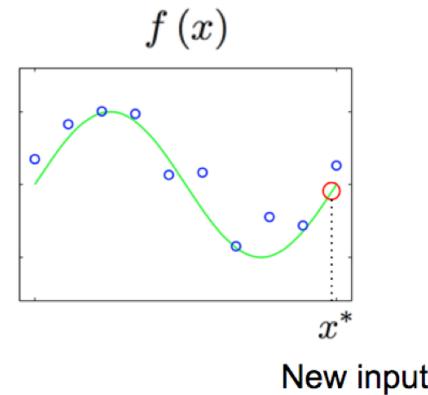
Make sense of the data

Given a dataset (X, y) consists of N rows(samples), we learn a function $f(x)$ that can predict unseen samples with minimal errors.

No.	Rating	Survey	Prize	Punishment	Aspen	Snowmass	Breckenridge	Keystone	Absin	Loveland	CrestedButte	Val	Silverton	WinterPark	MaryJane	Eldora
	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal	Nominal
1	0.675	20.0	10.0	30.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	1.0	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
3	0.9	20.0	10.0	30.0	1.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	1.0	1.0	0.0
4	0.95	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0
5	0.6	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0
6	0.95	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0
7	1.0	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
8	0.8	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	Q	1.0	1.0	1.0	1.0	1.0
9	0.9	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
10	0.85	20.0	10.0	30.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
11	0.94	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
12	1.0	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
13	0.8	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	Q	1.0	1.0	1.0	1.0	1.0
14	1.0	20.0	10.0	30.0	0.0	0.0	0.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
15	0.95	20.0	10.0	30.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0
16	0.9	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0
17	0.85	20.0	10.0	30.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
18	0.9	20.0	10.0	30.0	1.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	0.0
19	1.0	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
20	0.675	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.575	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0
22	0.925	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
23	0.9	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
24	0.6	20.0	10.0	30.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0

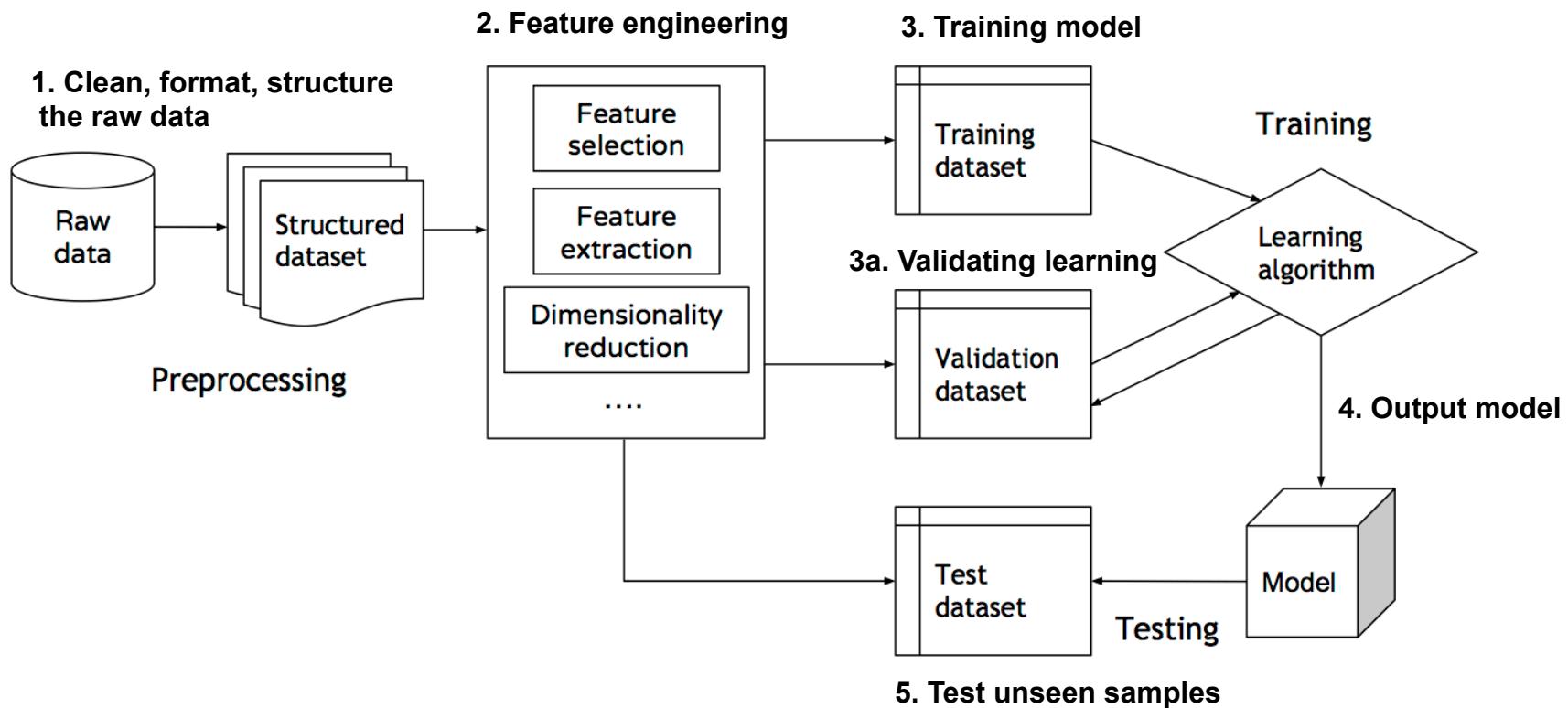


y (possibly not available)



New input

Learning Process

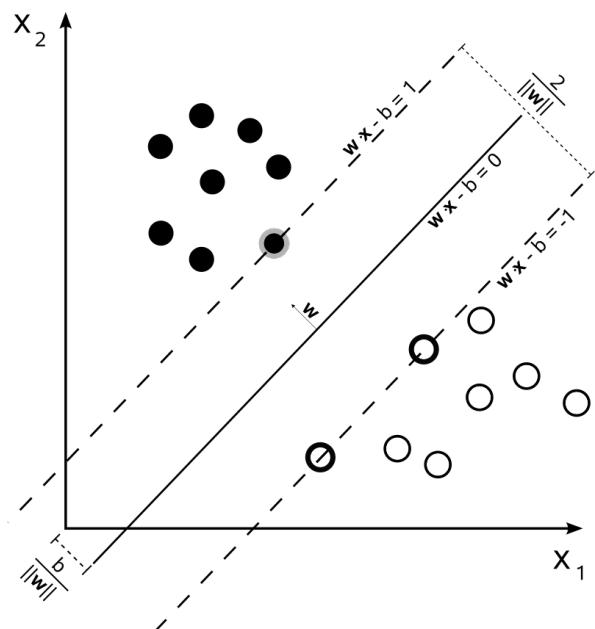


Learning Schema

Representation	Evaluation	Optimization
<p>Instance based</p> <ul style="list-style-type: none">• Support vectors• K-nearest neighbor <p>Tree based</p> <ul style="list-style-type: none">• Decision trees• Random forest <p>Graphical models</p> <ul style="list-style-type: none">• Bayesian networks• HMM	<ul style="list-style-type: none">• Accuracy or error rate• Precision and recall• Squared error• F1 score• KL divergence• Likelihood• Cross entropy	<ul style="list-style-type: none">• Greedy search• Gradient descent• Conjugate gradient• Newton method• Stoc. gradient descent• Linear programing• Quadratic programing

Example: SVM binary classification

Given N samples X and labels $y \in \{+1, -1\}$, find the maximal margin that separate both classes of samples.



Minimize the objective functions

$$\left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(w \cdot x_i + b)) \right] + \lambda \|w\|^2.$$

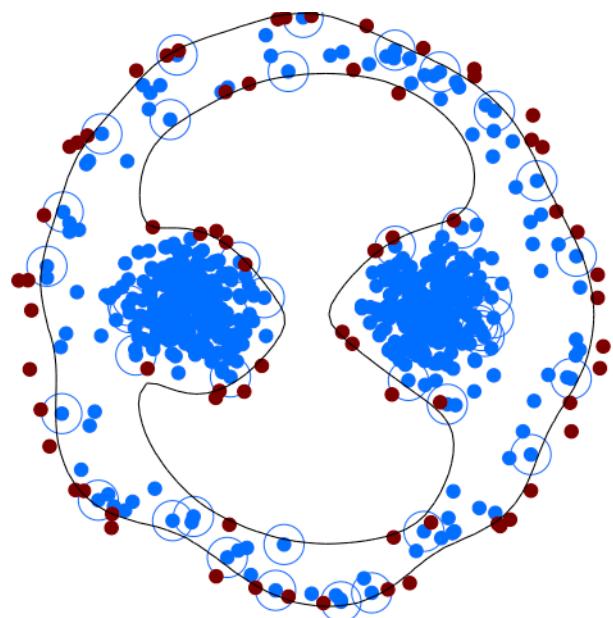
Hinge-loss Regularization

Effect of optimization

- Misclassified error is minimal
 - Separating hyperplane is not too complex

Example: One-class classification

Given N samples X and without labels, find an optimal data description of the majority class of the data samples. The rest is regarded as anomalies (outliers)



Learn a hypersphere of the data support

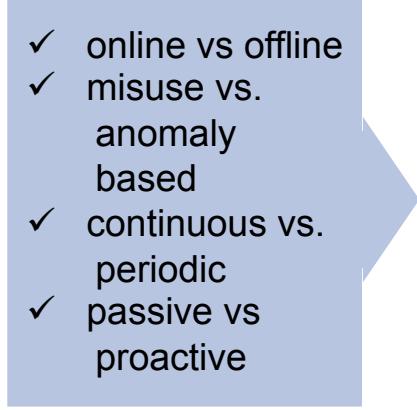
$$\begin{aligned} & \min_{R \in \mathbb{R}, \xi \in \mathbb{R}^\ell, c \in F} && R^2 + \frac{1}{\nu\ell} \sum_i \xi_i \\ & \text{subject to} && \|\Phi(\mathbf{x}_i) - c\|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0 \quad \text{for } i \in [\ell]. \end{aligned}$$

R: the radius of the hypersphere

Make Sense of the Security

Data sources

- ✓ Host based
 - ✓ system calls, sys logs
- ✓ Network-based
 - ✓ Inbound/outbound traffic
- ✓ Application logs
 - ✓ DB logs, web logs
- ✓ IDS sensor alerts
 - ✓ low level alarms
- ✓ Behaviour based
 - ✓ mouse, keystrokes
 - ✓ command lines
- ✓ Signatures databases

- 
- ✓ online vs offline
 - ✓ misuse vs. anomaly based
 - ✓ continuous vs. periodic
 - ✓ passive vs proactive

Make sense of it

- ✓ Detection
- ✓ Correlations
- ✓ Clusterings/Similarities
- ✓ Attack vectors
- ✓ Threat models
- ✓ Temporal/Spatial
- ✓ Prevention

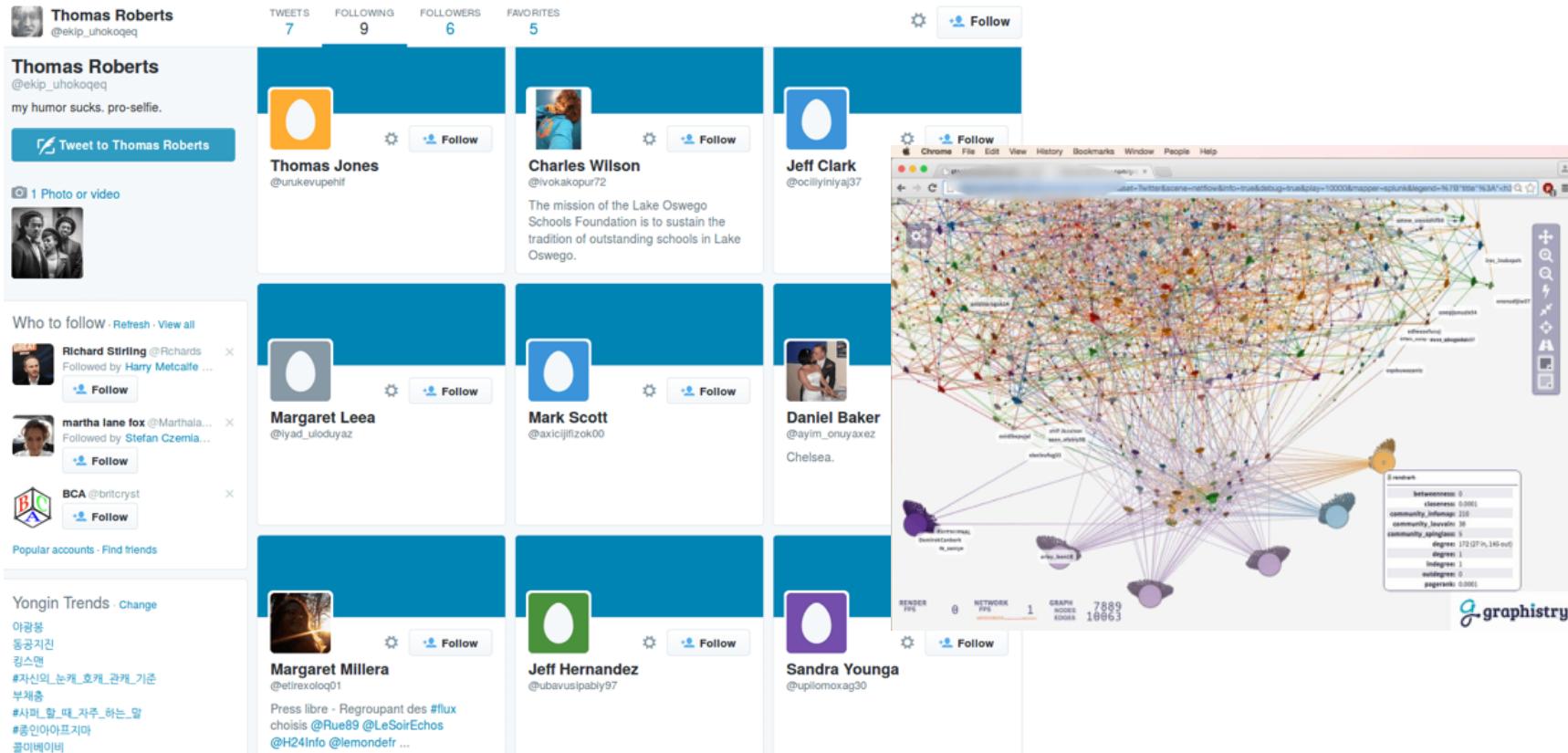
Outline

- Introduction & Motivation
- Machine Learning for Security
- **Research Subfields**
- Challenges & Discussion
- References

Botnets detection

- Networks of machines compromised by malware
- Threats:
 - Information/identity theft, DDoS, Software piracy, Spamming /Phishing...
- Very prevalent threats
 - It's easy, fun, and low risk ...
- Protocols: IRC, HTTP, IM, ...
- C&C: centralized, starred, hierarchical, random
- Fast fluxing, evolving, traffic encryption

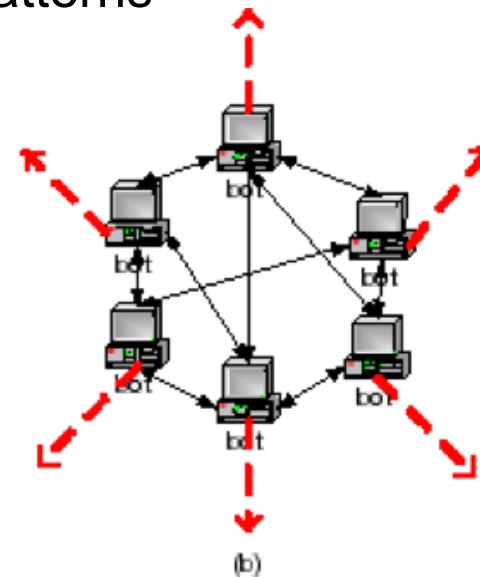
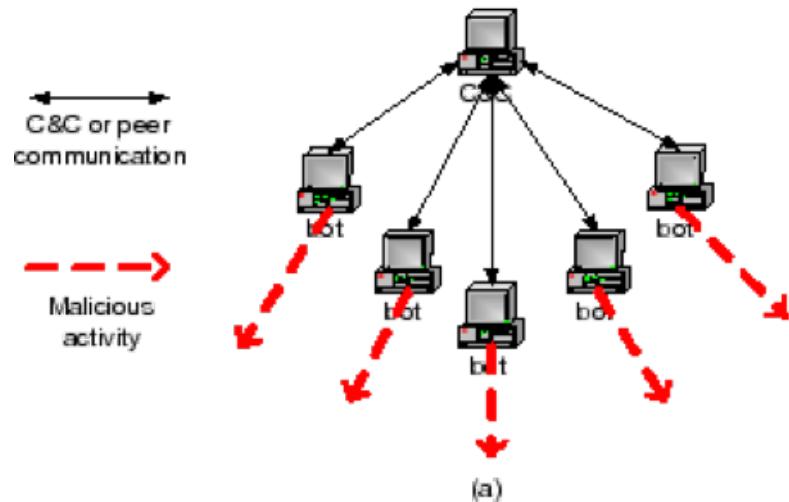
Botnets detection



BotMiner

Who is talking to whom?

C&C netflow patterns



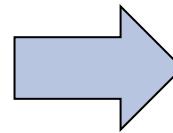
Activity patterns

Who is doing what?

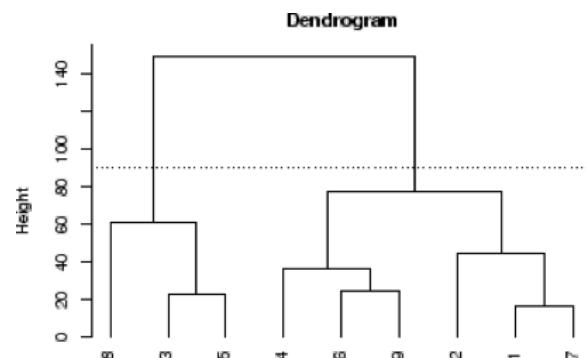
Guofei Gu, Roberto Perdisci, Junjie Zhang, and Wenke Lee. 2008. BotMiner: clustering analysis of network traffic for protocol- and structure-independent botnet detection. In Proceedings of the 17th conference on Security symposium (SS'08). USENIX Association, Berkeley, CA, USA, 139-154.

BotMiner

C-Plane features	e.g., ✓ #flows per hour (fph) ✓ #packets per flow (ppf) ✓ avg. #bytes per packets (bpp) ✓ avg. #bytes per second (bps)
A-Plane features	e.g., scanning, spam, binary downloading, exploit



hierarchical clustering



Output: clusters of machines
Bot or not?

Malicious Domains/URLs

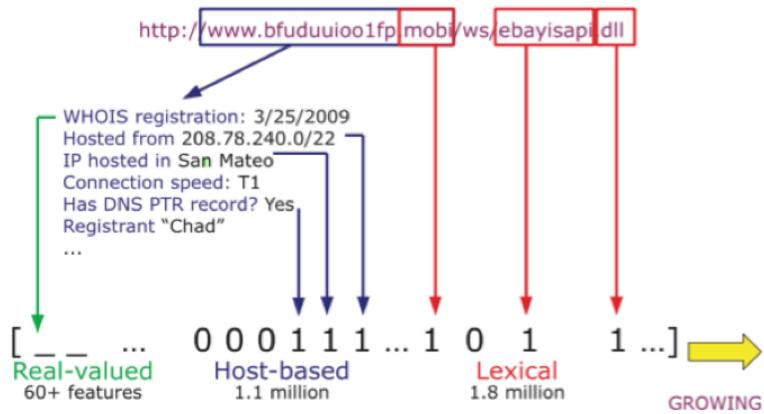
MALWARE DOMAIN LIST						
2016/03/30_12:54	importkauf.ch/45t3443r3	195.191.240.16	hos107.unaxus.net.	Locky ransomware	-	13030 
2016/03/30_12:54	buygrocery.nz/l7dsp	192.185.157.121	192-185-157-121.unifiedlayer.com.	Locky ransomware	Chitram TV NZ / info @chitramtv.nz	20013 
2016/03/30_12:54	creditfinancebank.ru/45t3443r3	92.53.112.82	armada.timeweb.ru.	Locky ransomware	-	9123 

Malware samples from www.malwaredomainlist.com

- Bots to locate C&C
- Links to phishing/spam/scam server
- Drive-by downloads

Learning malicious URLs

Data sources: 100 days, 20,000 URLs/day



Host-based
lexical-based

Features

- ✓ IP address properties
- ✓ WHOIS records
- ✓ Domain TTL
- ✓ Blacklist membership
- ✓ Geographic prop.
- ✓ Connections speed
- ✓ bag of words
- ✓ record every token
- ✓ Separate domain levels, host, path
- ✓ Sparse representation

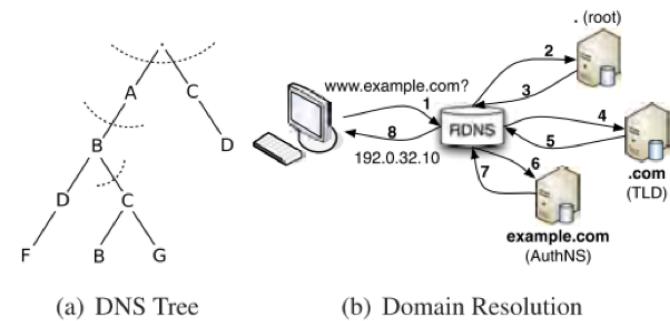
Online Classifiers

- ✓ Perceptron
- ✓ Stochastic gradient descent
- ✓ Confident weighted
- ✓ Passive aggressive

Ma, J., Saul, L.K., Savage, S. and Voelker, G.M., 2011. Learning to detect malicious URLs. ACM Transactions on Intelligent Systems and Technology, 2(3), pp.1–24.

Leveraging DNS traffic

Time-based	DNS answer based
<ul style="list-style-type: none"> ✓ short life ✓ daily similarity of requests ✓ repeating patterns ✓ access ratio 	<ul style="list-style-type: none"> ✓ #distinct IP addresses ✓ #distinct countries ✓ #domains IP shared ✓ Reverse DNS query results
TTL value-based	Domain name based
<ul style="list-style-type: none"> ✓ Avg. TTL ✓ Std. TTL ✓ #distinct TTL ✓ # TTL Change ✓ %usg. of TTL range 	<ul style="list-style-type: none"> ✓ %numerical characters ✓ %length of the longest meaningful substring



Features can be built from DNS traffic

Using C4.5 decision tree as classifier

- ✧ Bilge, L. et al., 2011. EXPOSURE : Finding Malicious Domains Using Passive DNS Analysis. Ndss., pp.1–17
- ✧ Antonakakis, M. et al., 2011. Detecting Malware Domains at the Upper DNS Hierarchy. USENIX Security Symposium., 11., pp.1–16.

Insider Attacks

- Traitor or Masquerader
 - Attacks from inside
- Threat: Inside > Outside
- Types of threats are diverse:
 - exfiltration of data, unauthorised downloads,
eavesdropping...
- Basic idea: user behaviour modelling
 - Unix command lines
 - Mouse movements, keystrokes
- Re-authentication, continuous authentication

Unix shell commands modeling

```
445 jekyll server --help
446 jekyll server -B
447 atom .
448 ps aux | grep 'jek'
449 kill 9 3530
450 kill -9 3530
451 kill 3530
452 ps aux | grep 'jek'
453 pwd
454 jekyll server -B
455 atom .
456 atom .
457 kill -9 3530
458 kill -9 11035
459 jekyll server
460 pwd
461 cd ~/repos/h3lib
462 git status
463 git add talks/ML4Sec/*
464 git commit -m 'add talk'
465 git push origin master
466 git branch
467 git checkout -b test
468 git add .
469 git commit -m 'initial commit'
470 git push origin test
```

Schonlau dataset

- ✓ 70 users
- ✓ 20 masquerade users
- ✓ 15,000 commands per user
- ✓ Block size: 100
- ✓ clean or dirty block

Bag of words for One-class SVM works well

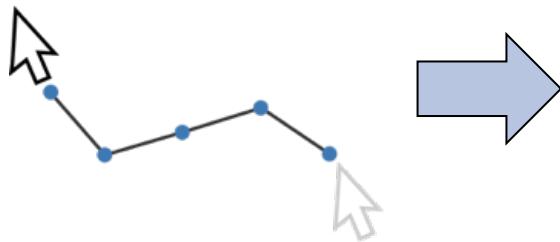
Wang, K. and Stolfo, S.J., 2003. One-class training for masquerade detection. Workshop on Data Mining for Computer Security, Melbourne, Florida., pp.10–19.

Behavior Biometrics

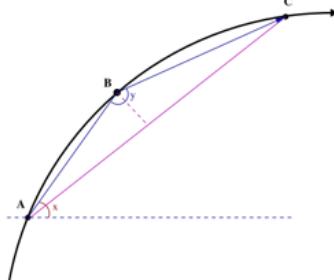
(physical) What you are	(behavior) What you do	To detect an insider...
<ul style="list-style-type: none">✓ Fingerprint✓ Retina pattern✓ Voice	<ul style="list-style-type: none">✓ Handwritten signature✓ Gait✓ Keystrokes✓ Mouse dynamics	<ul style="list-style-type: none">✓ Input interaction:<ul style="list-style-type: none">✓ mouse, keystrokes✓ Application level:<ul style="list-style-type: none">✓ GUI events✓ Habits, strategy✓ Low-level:<ul style="list-style-type: none">✓ System traces✓ Audit logs✓ Registry access✓ In/Outbound traffic

Mouse dynamics (cont. auth.)

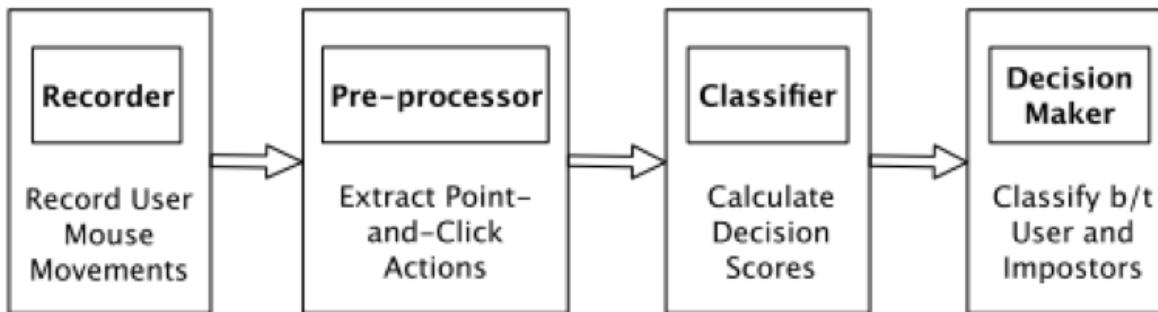
Point-and-click patterns



Angle-based features



Using binary SVM classifier

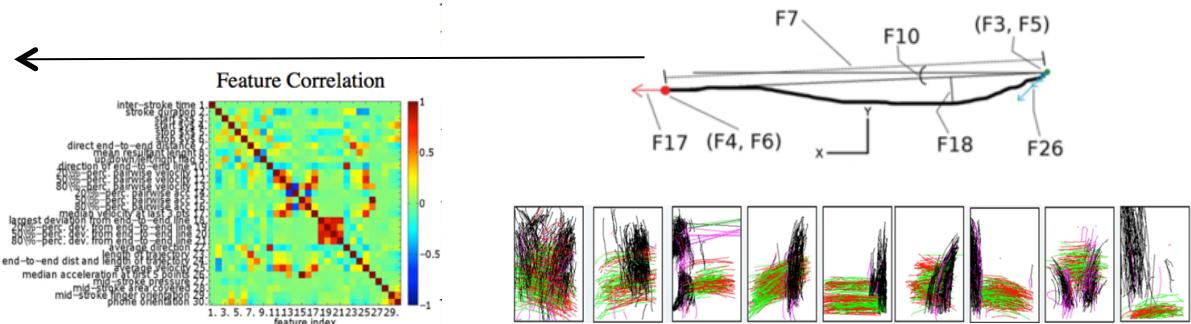


Zheng, N., Paloski, A. and Wang, H., 2011. An efficient user verification system via mouse movements. Proceedings of the 18th ACM conference on Computer and communications security - CCS '11., p.139

Touchalytics (cont. auth.)

Feature selection

Rel. mutual information	Feature description
20.58%	mid-stroke area covered
19.63%	20%-perc. pairwise velocity
17.28%	mid-stroke pressure
11.06%	direction of end-to-end line
10.32%	stop x
10.15%	start x
9.45%	average direction
9.43%	start y
8.84%	average velocity
8.61%	stop y
8.5%	stroke duration
8.27%	direct end-to-end distance
8.16%	length of trajectory
7.85%	80%-perc. pairwise velocity
7.24%	median velocity at last 3 pts
7.22%	50%-perc. pairwise velocity
7.07%	20%-perc. pairwise acc
6.29%	ratio end-to-end dist and length of trajectory
6.08%	largest deviation from end-to-end line
5.96%	80%-perc. pairwise acc
5.82%	mean resultant lenght
5.42%	median acceleration at first 5 points
5.39%	50%-perc. dev. from end-to-end line
5.3%	inter-stroke time
5.14%	80%-perc. dev. from end-to-end line
5.04%	20%-perc. dev. from end-to-end line
5.04%	50%-perc. pairwise acc
3.44%	phone orientation
3.08%	mid-stroke finger orientation
0.97%	up/down/left/right flag
0%	change of finger orientation



Frank, M. et al., 2013. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Transactions on Information Forensics and Security, 8(1), pp.136–148.

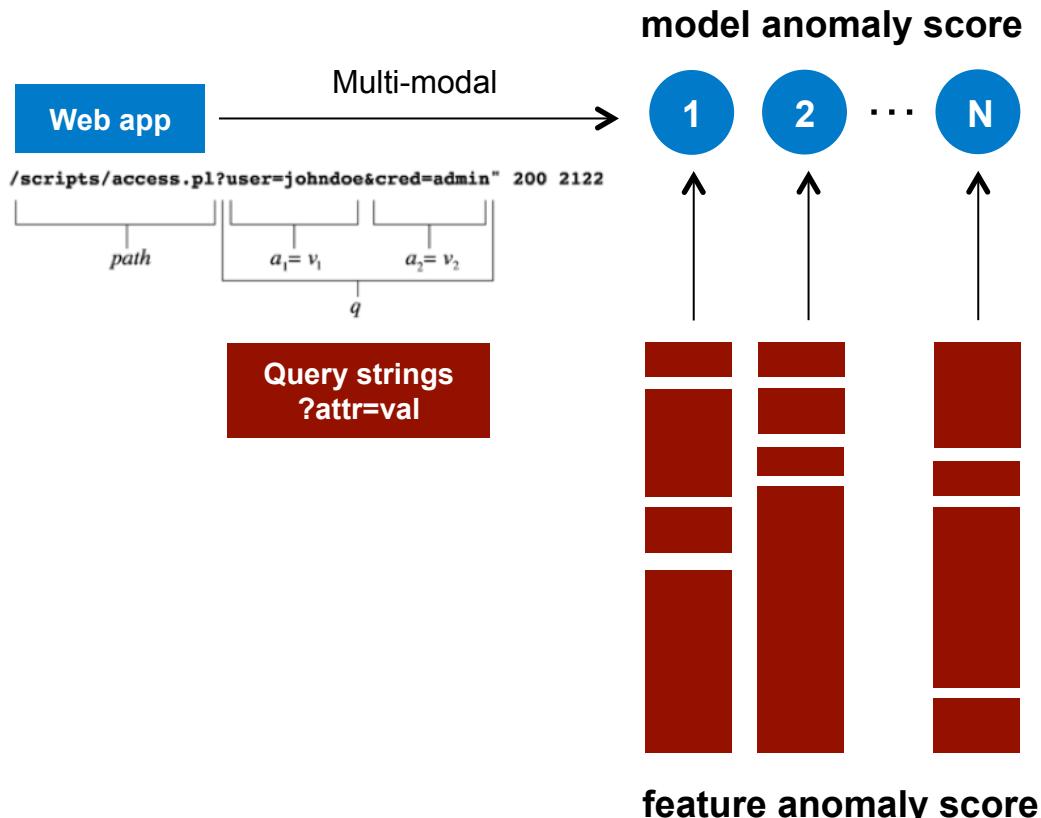
Web Attacks

- Attacks on web applications
- Types: SQLi, RFI, LFI, XSS, DT, CS, etc...
- Web Server Logs: conform to CLF (Common Log Format)
 - *remotehost rfc931 authuser [date] "request" status bytes*
- Misuse-based vs. Anomaly-based
 - Misuse: match signature databases
 - Anomaly: able to detect unknown patterns
- Application oriented multi-modal IDS

Web Attacks

Example (log entry):

```
169.229.60.105 - john Doe [6/Nov/2002:23:59:59 -0800] "GET /scripts/access.pl?user=john Doe&cred=admin" 200 2122
```



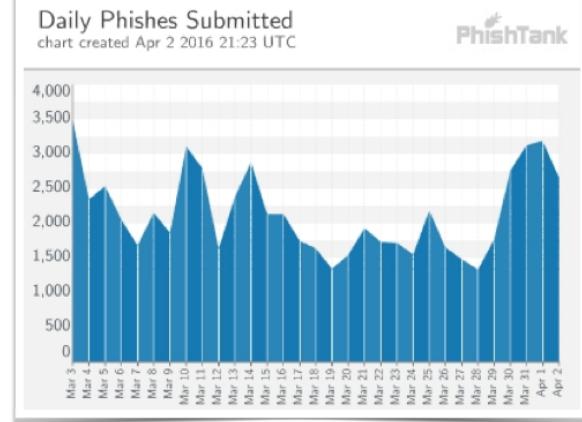
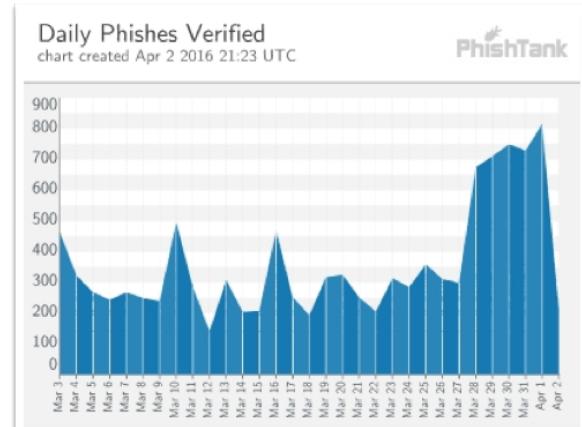
Features

- ✓ attribute length
- ✓ attribute grammar
- ✓ attribute presence
- ✓ attribute order
- ✓ character distribution
- ✓ requests time delay
- ✓ token finder
- ✓ invocation order

Kruegel, C., Vigna, G. and Robertson, W., 2005. A multi-model approach to the detection of web-based attacks. Computer Networks, 48(5), pp.717–738.

Phishing/Spam

- Normally legitimate-like website/Email
- Detection tools:
 - Spoofguard, Netcraft, SpamAssassin ...
- Domain/Content based features



source: PhishTank

Phishing/Spam

Features

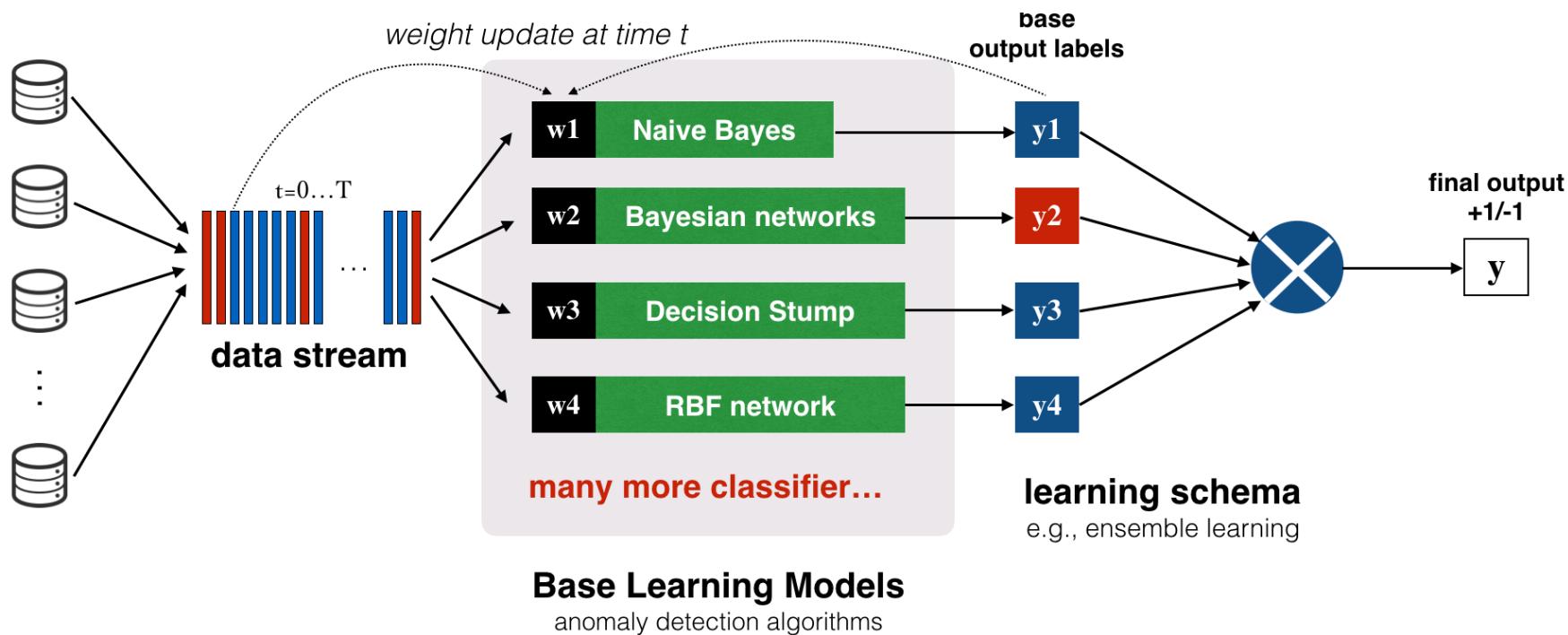
- | | |
|-----------------------------|---|
| ✓ IP-based URL | ✓ http://192.168.0.1/paypal.cgi?fix account |
| ✓ linked-to domain age | ✓ less than 60 days? |
| ✓ href/text of link matched | ✓ paypal.com |
| ✓ “here” link | ✓ links to other domain |
| ✓ Number of links | ✓ links by <a> tags with href attribute |
| ✓ Number of domains | ✓ main part of domain name counts |
| ✓ Number of dots | ✓ more dots is suspicious |
| ✓ If contains javascript | ✓ find text ‘javascript’ |
| ✓ Spam filter output | ✓ result by spam detector |

Fette, I., Sadeh, N. and Tomasic, A., 2007. Learning to detect phishing emails. Proceedings of the 16th international conference on World Wide Web - WWW '07., p.649

Adaptive IDS

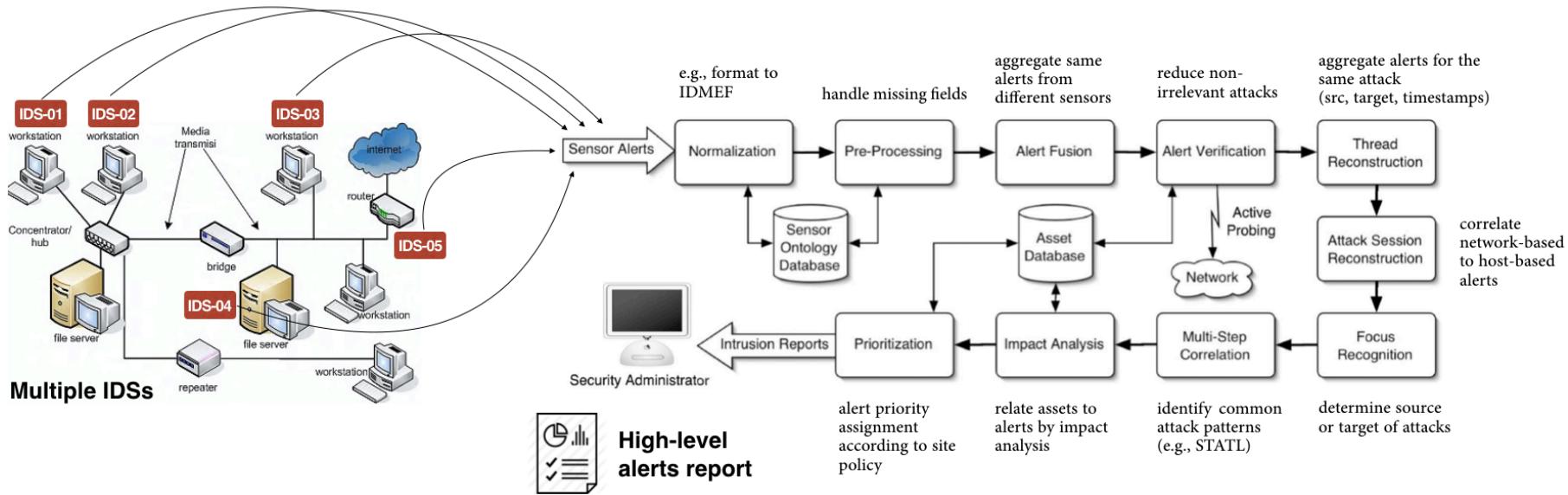
- Diverse network environments
- Possibly adversarial environment
- Single modal IDS not robust
- Dynamic attack landscape
- Heterogeneous data sources (no free lunch)
- Multiple Classifier System (MCS)
- **Goal:** adaptive to changes

Adaptive IDS



Nguyen, H.T. and Franke, K., 2012. Adaptive Intrusion Detection System via online machine learning. Proceedings of the 2012 12th International Conference on Hybrid Intelligent Systems, HIS 2012., pp.271–277.

Alert Correlation

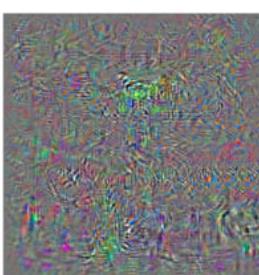


F. Valeur, G. Vigna, C. Kruegel and R. A. Kemmerer, "Comprehensive approach to intrusion detection alert correlation," in IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 3, pp. 146-169, July-Sept. 2004.

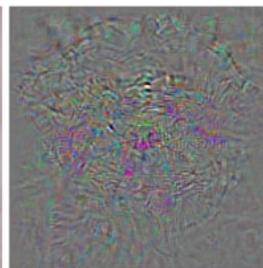
Adversarial Learning

- What if a bot learns to behave like a legitimate user?
- Learning algorithm itself can also be a security target
- Data stationary assumption is broken
- Attacks: Causative or Evasive
- Passive or proactive arm-races
- Aim to design learning algorithms robust to adversaries

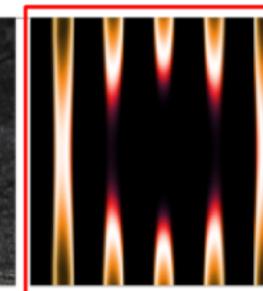
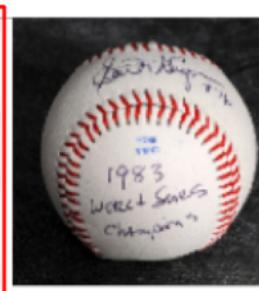
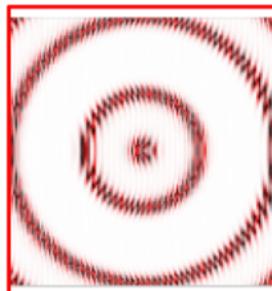
Adversarial Learning



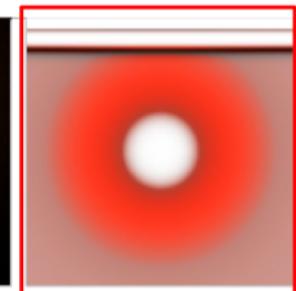
School bus + *perturbation* = Not a school bus



Puppy + *perturbation* = Not a puppy



Baseball (99%)



Matchstick (99%)

Pingpong ball (99%)

Deep net is easily fooled

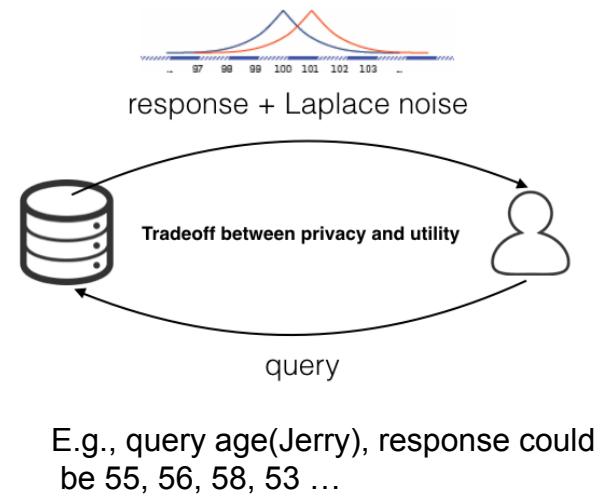
Nguyen, A., Yosinski, J. and Clune, J., 2014. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In Computer Vision and Pattern Recognition 2015. Boston, Massachusetts, US.

Differential Privacy

- Big data is booming up, but privacy is less considered
- Sensitive information can be inferred from statistics
- **Differential privacy:** even only aggregated statistics is published from samples, we can never tell/infer anything about a particular user.
- Examples:
 - AOL search results scandal
 - Facebook ads -> unclose private information
 - Netflix competition

Differential Privacy

Name	Age	Smoke	Lung cancer
Ian Jonathan	57	Yes	Positive
Kelly Nyn	60	Yes	Positive
Christ Lee	48	No	Negative
Tim Kingston	68	No	Negative
Mark Spring	35	Yes	Negative
Liu Han	78	No	Positive
Yoshua Hiro	40	Yes	Negative



Statistics: $\text{Prob}(\text{lung cancer})=1 \text{ where } \text{age}>50 \text{ & smoke=Yes}$

What if Jerry, age=55, smoke=Yes, ... privacy leak

Dwork, C., 2008. Differential Privacy: A Survey of Results. In Theory and Applications of Models of Computation. pp. 1–19., pp.1–19

Outline

- Introduction & Motivation
- Machine Learning for Security
- Research Subfields
- Challenges & Discussion
- References

Challenges

- Outlier detection is hard
- High cost of errors (false alarm)
- Lack of appropriate training data
 - Anomalous class is rare
- Interpretation of results
 - How to explain the attacks?
- Variability in network traffic
 - Attack patterns evolve
- Adaptive adversaries
- Evaluation difficulties

References

1. Emilio Ferrara, Onur Varol, Clayton A. Davis, Filippo Menczer, Alessandro Flammini: The Rise of Social Bots. CoRR abs/1407.5225 (2014)
2. Guofei Gu, Roberto Perdisci, Junjie Zhang, and Wenke Lee. 2008. BotMiner: clustering analysis of network traffic for protocol- and structure-independent botnet detection. In Proceedings of the 17th conference on Security symposium (SS'08). USENIX Association, Berkeley, CA, USA, 139–154.
3. Ma, J., Saul, L.K., Savage, S. and Voelker, G.M., 2011. Learning to detect malicious URLs. ACM Transactions on Intelligent Systems and Technology, 2(3), pp.1–24.
4. Bilge, L. et al., 2011. EXPOSURE : Finding Malicious Domains Using Passive DNS Analysis. Ndss., pp.1–17
5. Antonakakis, M. et al., 2011. Detecting Malware Domains at the Upper DNS Hierarchy. USENIX Security Symposium., 11., pp.1–16.
6. Wang, K. and Stolfo, S.J., 2003. One-class training for masquerade detection. Workshop on Data Mining for Computer Security, Melbourne, Florida., pp.10–19.
7. Zheng, N., Paloski, A. and Wang, H., 2011. An efficient user verification system via mouse movements. Proceedings of the 18th ACM conference on Computer and communications security - CCS '11., p.139
8. Frank, M. et al., 2013. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Transactions on Information Forensics and Security, 8(1), pp.136–148.
9. Kruegel, C., Vigna, G. and Robertson, W., 2005. A multi-model approach to the detection of web-based attacks. Computer Networks, 48(5), pp .717–738.
10. Fette, I., Sadeh, N. and Tomasic, A., 2007. Learning to detect phishing emails. Proceedings of the 16th international conference on World Wide Web - WWW '07., p.649
11. Nguyen, H.T. and Franke, K., 2012. Adaptive Intrusion Detection System via online machine learning. Proceedings of the 2012 12th International Conference on Hybrid Intelligent Systems, HIS 2012., pp.271–277.
12. F. Valeur, G. Vigna, C. Kruegel and R. A. Kemmerer, "Comprehensive approach to intrusion detection alert correlation," in IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 3, pp. 146-169, July-Sept. 2004.
13. Nguyen, A., Yosinski, J. and Clune, J., 2014. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In Computer Vision and Pattern Recognition 2015. Boston, Massachusetts, US.
14. Xiao, H. et al., 2015. Is Feature Selection Secure against Training Data Poisoning ? Int'l Conf. on Machine Learning (ICML), 37.
15. Xiao, H. et al., 2014. Support Vector Machines under Adversarial Label Contamination. Journal of Neurocomputing, Special Issue on Advances in Learning with Label Noise, 160(0), pp.53–62.
16. Dwork, C., 2008. Differential Privacy: A Survey of Results. In Theory and Applications of Models of Computation. pp. 1–19., pp.1–19