Preventing botnets

- Traditional
  - code analysis and finding malware fingerprints

- Code/binary analysis is mostly manual and increasingly harder
  - Code obfuscation
  - Encryption
  - Self-modifying

- Behavior-based analysis is much harder to thwart
  - Bots need to communicate!
Preventing botnets

- Traditional
  - code analysis and finding malware fingerprints
- Code/binary analysis is mostly manual and increasingly harder
  - Code obfuscation
  - Encryption
  - Self-modifying
- Behavior-based analysis is much harder to thwart
  - Bots need to communicate!

Lots of data available...
Binary code (Zeus)
Network traffic (HLUX2)
Classic Approach:

1. take a huge data set
2. compute features
3. train a classifier
4. deploy the classifier on test

Machine learning

- train data
- learning algorithm
- model
- test data
- output
The colors indicate the type of model, from left to right: **logical**, probabilistic and **geometric**.

- **Logical** models include: association rules, trees & rules, Naïve Bayes, and mixture models.
- **Probabilistic** models include: support vector machines, linear classifiers, and k-means.
- **Geometric** models include: k-means and nearest neighbour.

Based on Peter Flach
Classic ML fails in cyber security

- Large majority (> 99%) of cases are benign!
  - adapt data/models, otherwise no positives

- Data is massive and keeps coming in!
  - need to count quickly, reduce false positives

- There is an opponent, they learn too!
  - avoid using generic fingerprints/simple rules

- Privacy makes data inaccessible...
Bloom filter: dealing with massive data

http://www.jasondavies.com/bloomfilter/

Key: Add

Definitely not there.
Course

- Read scientific papers, use techniques on real data:
  - Credit fraud data from Adyen
  - Botnet traffic from HLUX2
  - NetFlow traces, perhaps from EEMCS…
  - ...

- Learn by doing in labs, an exam on content