Catching DNS tunnels with ~~IDS that doesn’t suck~~ A.I.

A talk about Artificial Intelligence, geometry and malicious network traffic.

Agenda

- Introduction
- Neural Network basics
- DNS Tunnel Basics
- Data mining DNS tunnels out of network traffic with a Neural Network.
- Destination… unknown…

Introduction

- The goal of the project was to reliably discover DNS tunnels out of network traffic.
- Hopefully, you’ll learn a lot from it. If nothing else, maybe it will inspire someone to do something interesting.

A.I.

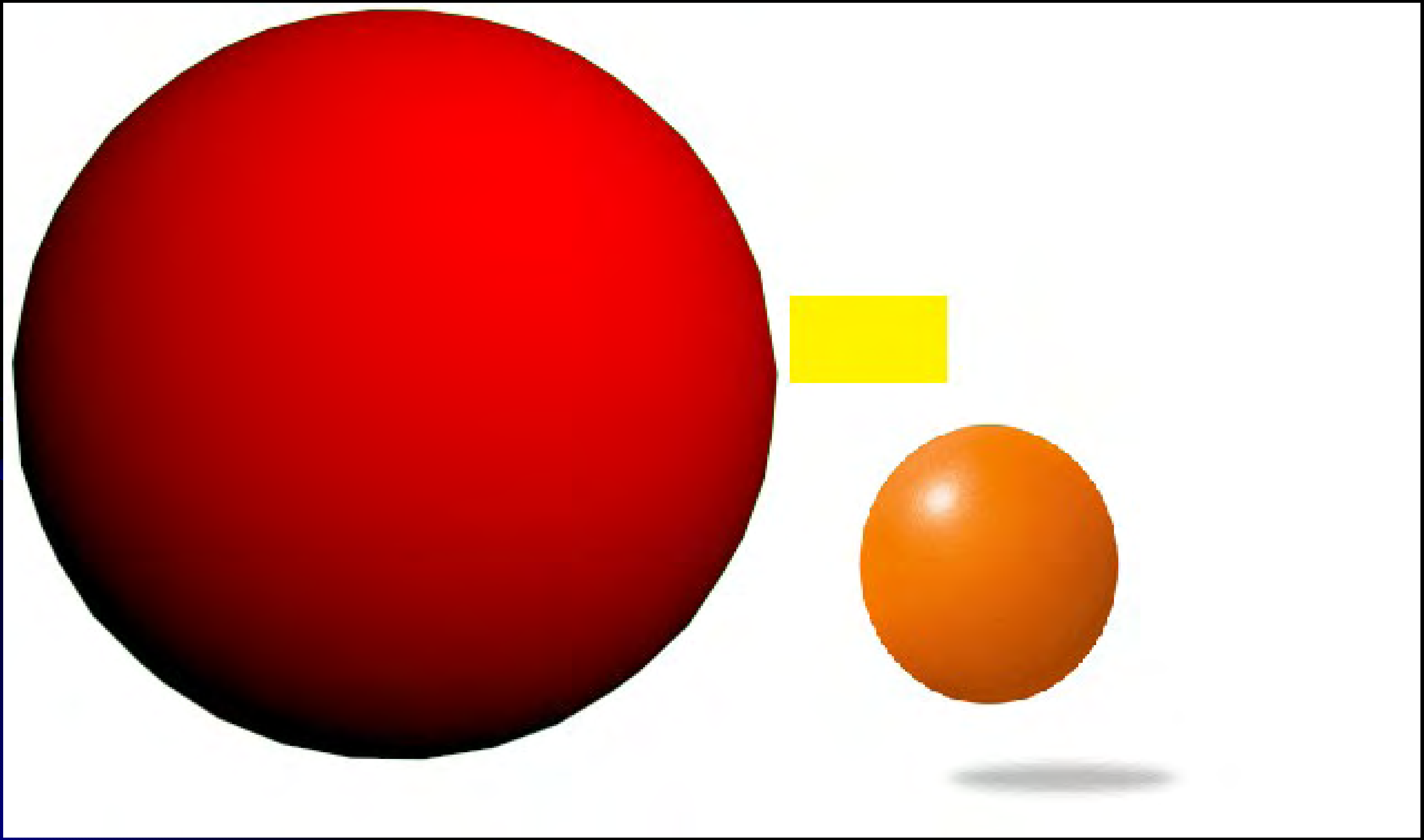
- When I say A.I., most people start thinking of a computer with personality or traits, movie or book computer characters, replacing their spouse with a robot etc.

- Not really what *I* am talking about when I say AI. How about a program that gets a computer to make a difficult distinction or decision.

Classification

- Classification is one such distinction or decision.
- So, lets get a more clear discussion of AI by talking about how we think.
- Or how we think we think, we think.
- Ugh.

Which is the apple, orange and banana?



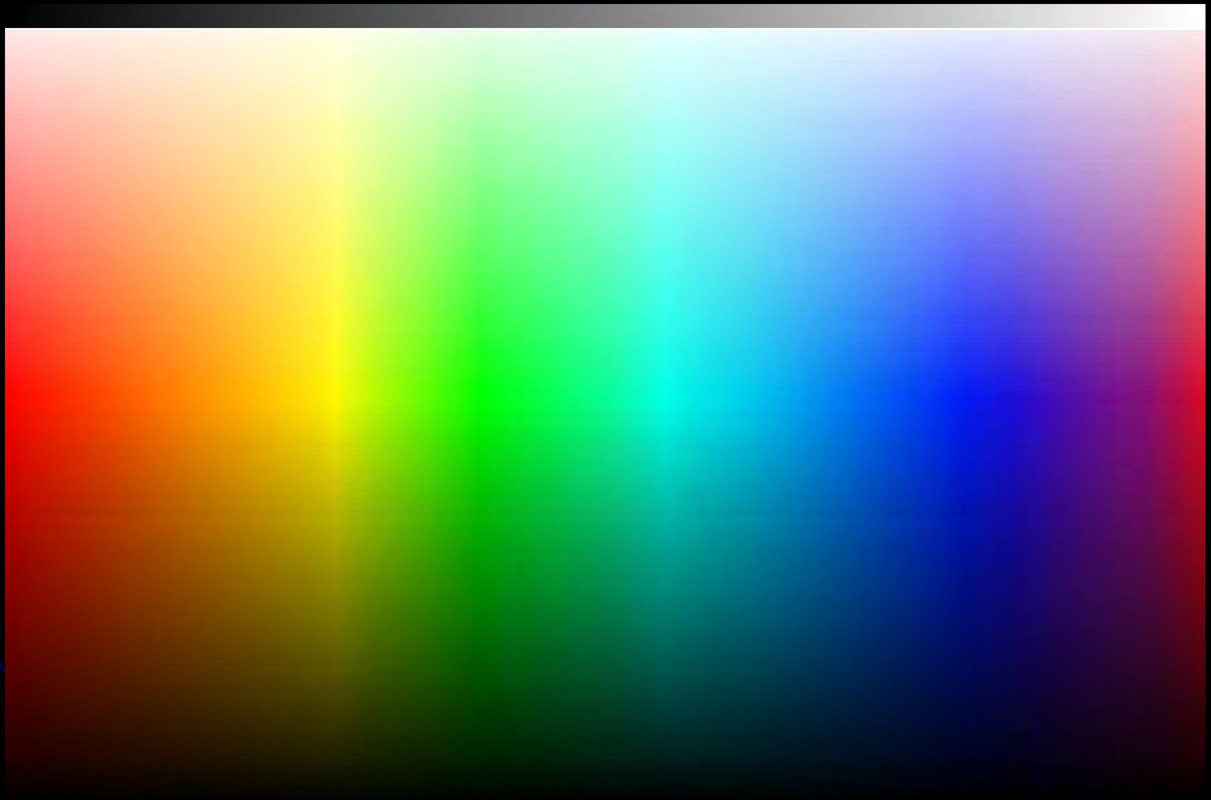
What did your brain just do?

- We made a classification on abstract objects based on traits of real objects.

- You pulled *traits* from real life.
- You assigned *weights* to those traits.

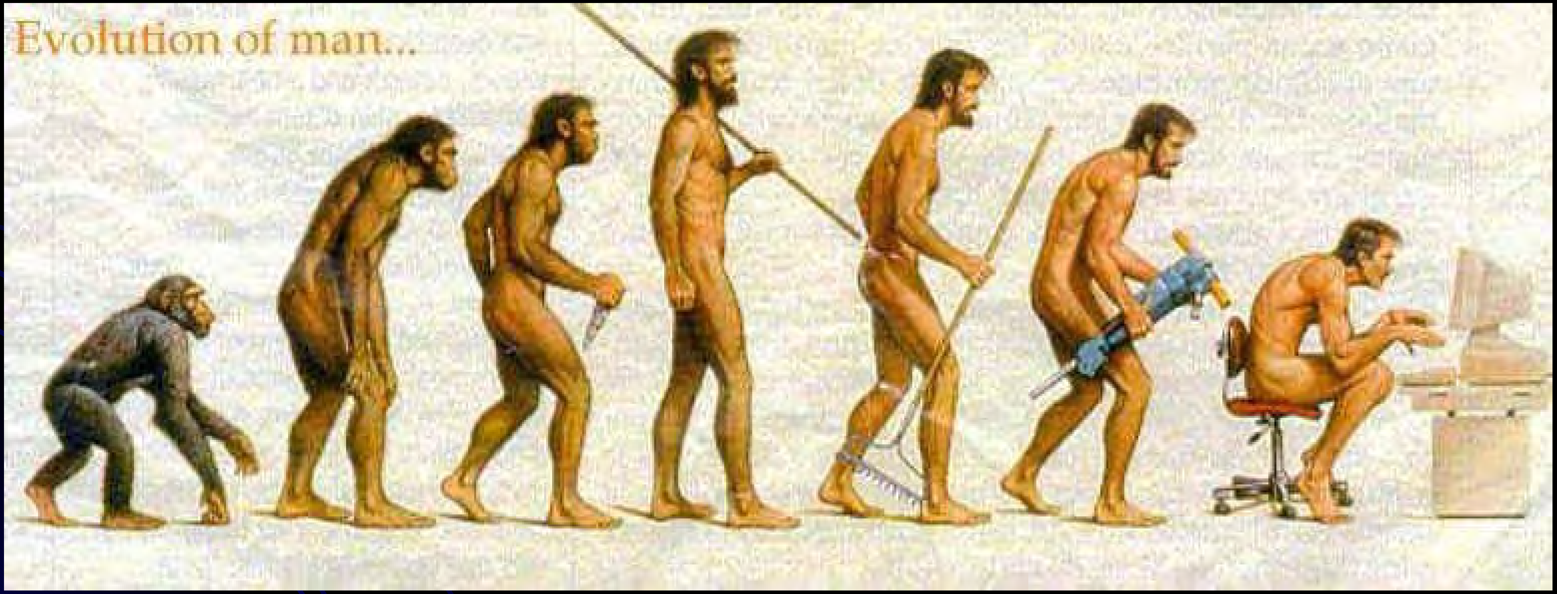
Easy with only a few traits…so you think. Enter thresholds…

- When does the following become green or blue?



What if we have LOTS of traits involved?

Remember, we need traits we can measure.



Enter Artificial Neural Nets

- Key terms explain it all.

- “non-linear statistical data modeling”
- “adaptive”
- “can be used to model complex relationships between inputs and outputs or to find patterns in data”

Don’t worry, it’s easy

- We have software that reproduces what we just did.
- We are taking a bunch in inputs (traits)
- We give them values (assign weights)
- We ADAPT our decisions until they match our training data. (set thresholds)
- If you have an answer cheat-sheet, its called supervised learning.

ANN

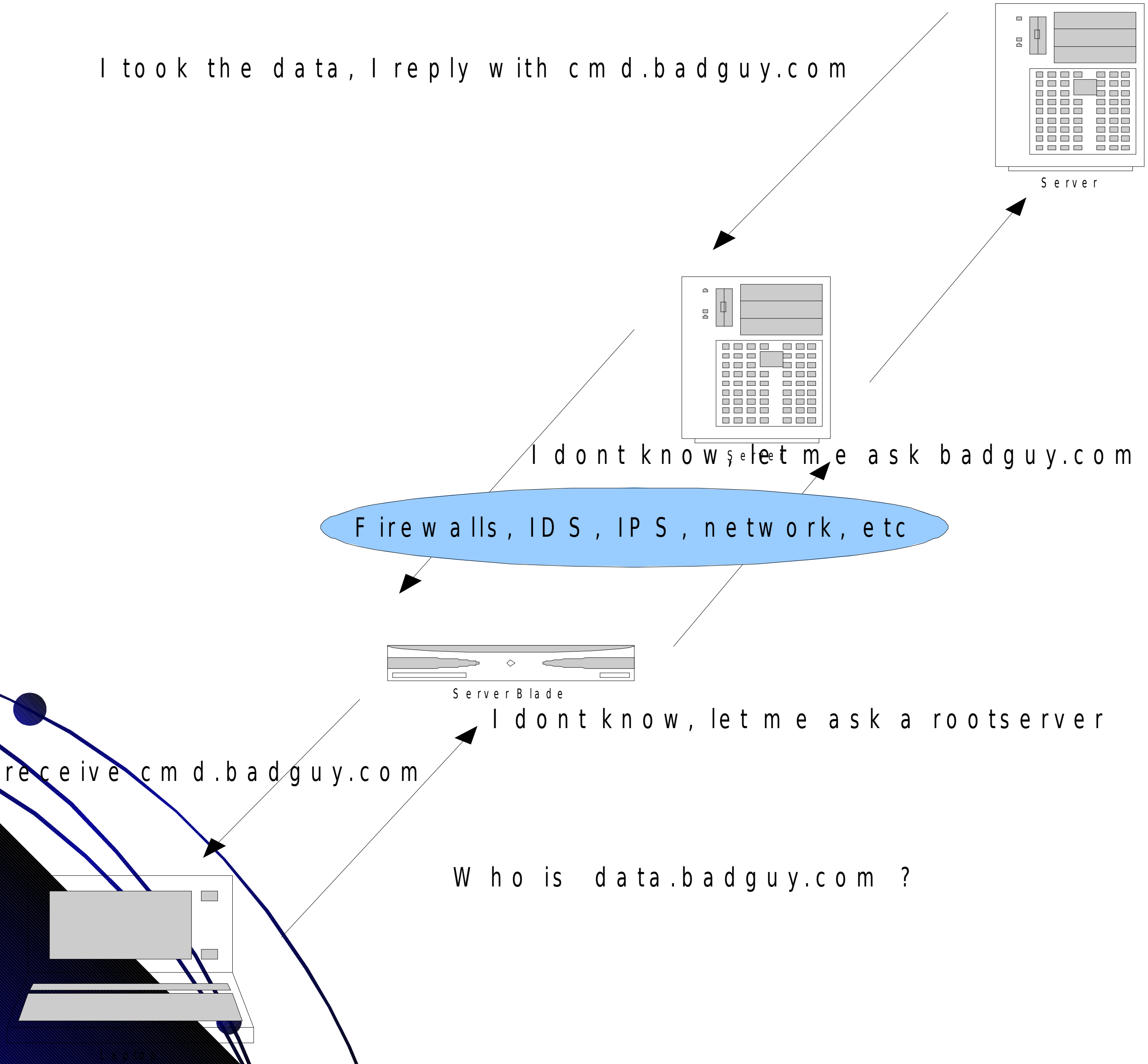
- I’d rather not get into the nuts and bolts of how the ANN’s work, unless you have questions.
- And keep this in mind, you don’t need to know how they work.
  1. Build the problem, define the decision, select the traits, assign weights.
  2. USE SOMEONE ELSES ANN PACKAGE.
  3. You only need to know the ins and outs during the final stage, tuning the ANN.

Agenda

- ~~Introduction~~
- ~~Neural Network basics~~
- DNS Tunnel Basics
- Data mining DNS tunnels out of network traffic with a Neural Network.
- Destination… unknown…

What is a DNS tunnel?

- DNS is Domain Name System
- Used, in general, to map IP addresses to domain names. (yes, I know, has lots of other uses)
- DNS tunnels are so lethal because they work from nearly anywhere.



I took the data, I reply with cmd.badguy.com

I dont know, let me ask badguy.com

Firewalls, IDS, IPS, network, etc

I dont know, let me ask a rootserver

I receive cmd.badguy.com

Who is data.badguy.com ?

The key points are this:

- DNS is an automatic route out of a network and to the malicious host, if the data is in the request or response.
- DNS requests that are not cached get routed to an authoritative server for that domain.

Make it a tunnel

- So if I make a request to data2exfiltrate.diaboloicalplans.com
- It will be eventually “routed” in the DNS protocol to diabolicalplans.com DNS server.
- The DNS server will strip off the data, and respond with either a command.diabolicalplans.com or an IP address.
- I can run a complete command and control tunnel for a trojan in this fashion. I can exfiltrate as much data as I want, you cant stop the signal (of DNS).

Okay, that’s a tunnel. Here’s some takeaway

- Historical IDS and now IPS are primarily concerned with detecting or stopping attacks.
- This is pretty useless because it’s too hard.
- **Hunting for egress C&C, or tunnels, traffic is a better way to catch intruders.**
- “If you get in, you have to get out.”

Agenda

- ~~Introduction~~
- ~~Neural Network basics~~
- ~~DNS Tunnel Basics~~
- Data mining DNS tunnels out of network traffic with a Neural Network.
- Destination… unknown…

So, first things first.

- Why an ANN to look for DNS tunnels?
  - Turns signatures away from packets, into traits, weights and thresholds. 2 of the three things there we don’t even set, the ANN does during its “learning” phase.
  - But mostly because of their adaptive abilities. This allows me to be even lazier…

Cont.

- The ANN will use the method we just learned to look at DNS traffic.
- If the weights or thresholds are set to low, and we find a DNS tunnel we cant identify, we just add it to our training data and “re-learn”.
- Learning allows the ANN to reset new weights and thresholds (not traits) to find this unknown tunnel.
- We should NEVER have to rewrite another signature by hand.

Cont.

- Snort, <edit> and <edit> all pretty much suck (currently) at finding DNS tunnels. I’m not familiar with others.
- I found lots of references on the web to using ANN’s and statistics to find DNS tunnels, but I couldn’t find an actual packaged idea.

Step 1: Frame the Question Correctly

- Most AI projects don’t do this.
- We have to get a classification or decision that is simple. You can add multiple simple decisions together to make a bigger one if needed, but we don’t need that.
- Ours is, “Are requests to a domain part of a tunnel or not”
- Okay, now we need to go get traits.

Traits

1. We will track each domain by its name.
2. How many packets to that domain?
3. Average length of packets to that domain?
4. Average number of distinct characters in the lowest level domain.
5. And… hhmmm…

Something is missing

“I was told there’d be no math”

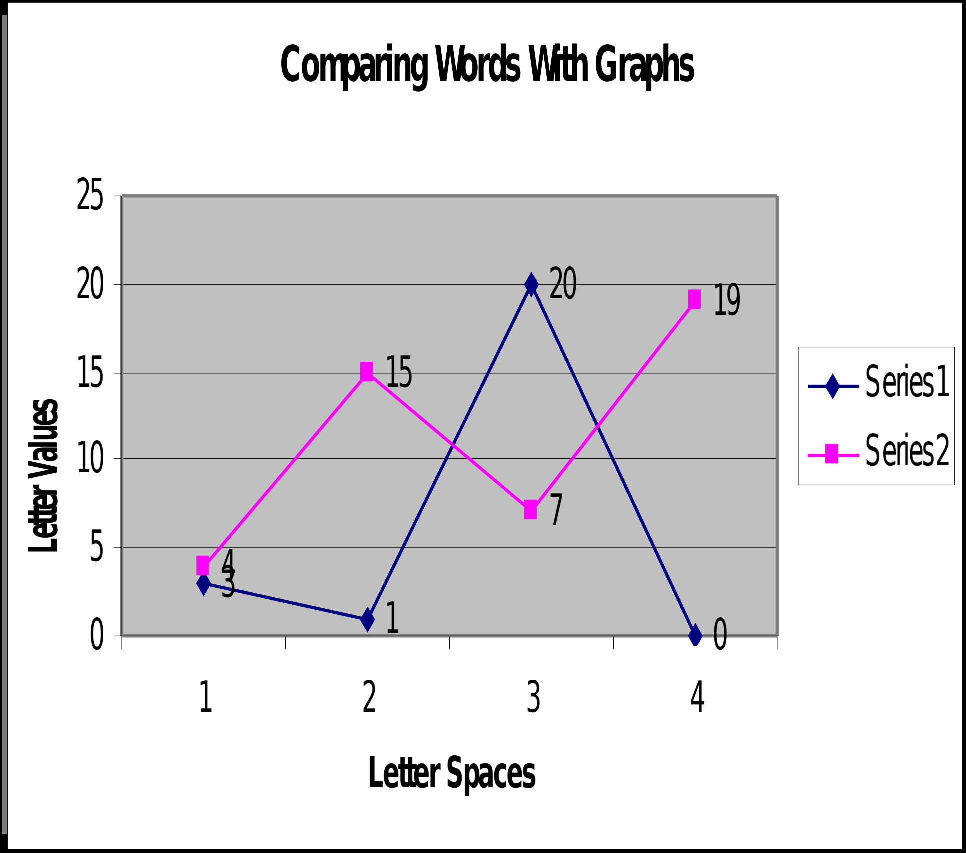
- I originally planned to capture the “entropy” or “information gain” in each LLD (aka Shannon’s theorems). (LLD == Lowest Level Domain)
- But this doesn’t work.
  - Lkwoeiurhdan.diabolicalplans.com
  - It has a high “entropy” as opposed to www, but Hostname? foreign language? Encoded data?
  - If I see it in 16 requests, then I can probably make an assessment (or an educated guess).

“I was told there’d be no math”

- So, what I *REALLY* wanted was a way to compare LLD’s in the same domain to each other.
- How much is LLD in request 1, like LLD in request 2, like LLD in request 3, etc.
- If data is moving out of a tunnel via the LLD, the LLD’s will change a great deal (relative to their encoding).

“I was told there’d be no math”

- So, lets not think of LLD’s as strings.
- Let us think of them instead as structures.
- Lets look at an easy example. Dogs and cat in 2 dimensional space.



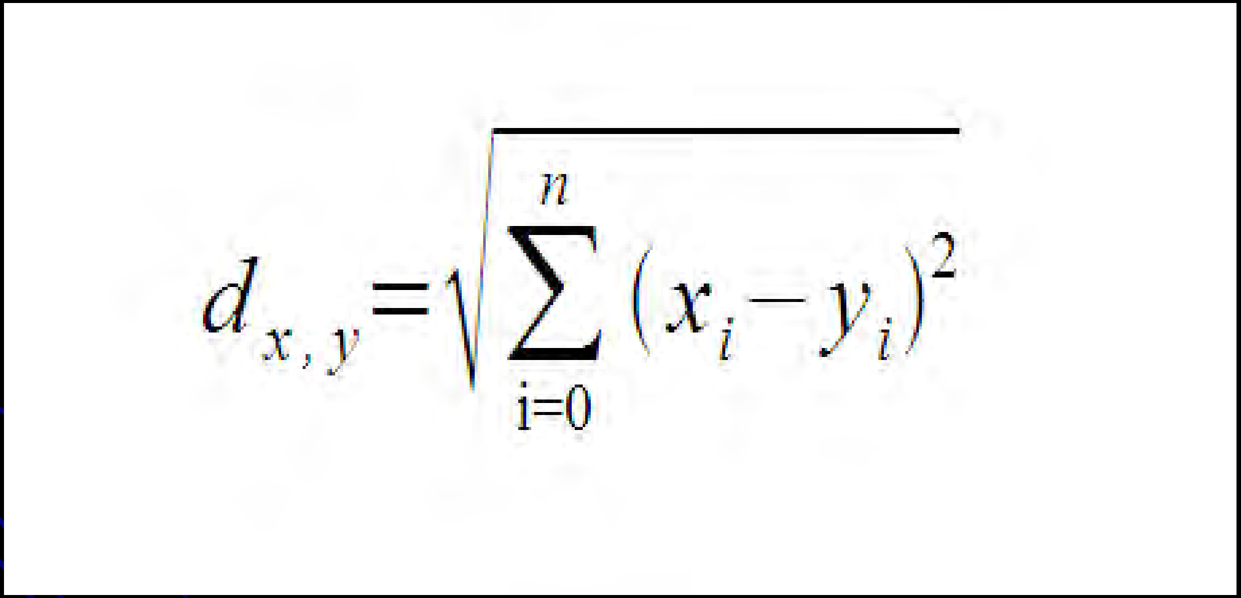
“I was told there’d be no math”

- So, what we will do is more complex than that. We need to normalize the data so that we can measure geographic distance.
- LLD’s can only have a limited number of chars in them, per RFC 1035.
- So lets think of each spot in an LLD as having 36 possible values [a-z + 0-9] and a NULL value for everything else.
- Now we have multi ordinal vectors… 8 chars means 8 dimensions…

“I was told there’d be no math”

- X = (r – 1)/(R – 1)
- So if we have 0 (null),A-Z,0-9
- X for A = (2 – 1)/ (36 – 1) = .0285
- Repeat until all characters are normalized.

Ordinal or Geometric Distance - **Normalized Rank Transformation**



$$d_{x,y}=\sqrt{\sum_{i=0}^{n}(x_i-y_i)^2}$$

So…uhhhh…

- So for each letter in the two LLD’s, we calculate a normalized value (which maybe null, which is 0) between 0 and 1.
- We sum the squared subtraction of each letter, and take the square root of that.
- This allows us to calculate the DISTANCE between LLD’s.

The Power of Cheese

- In Euclidian geometry another word for distance is SIMILARITY (or its inverse DIS-SIMILARITY).
- We are now able to calculate how much alike two LLD’s are.
- Is LLD 1 like LLD 2 ? How different are they? Are they different than LLD 3 ? How much ?
- Do you see the power of what we are now able to feed the Neural Net ?

Traits

- So now we have a pretty good list of traits.
- We now “train” the neural network using data we have control over.
- Then run it on real data, see what it finds.
- Anytime we have a false negative, we add the new data to the training list and retrain.
- Over-fitting and under-fitting are a concern. Go rent a real AI guy? Or

DNStTrap 0.9 FAQ

- Why version 0.9?
  - Its not iron clad, armored software. **It is POC only.**
  - Doesn’t sniff off the wire (windowing issues), uses pcap files instead
  - Real AI guys can probably tune the NN way better

- What are the major functions?
  - Findtunnels – looks at bulk data to find new tunnels
  - Newdata – creates a new training file entry
  - Train – train or retrain the NN

DEMO (or at end, depending on time)

How Does it Work?

- Well, it works.
- Caught the following without tuning:
  - Iodine
  - Ozzyman
  - Dns2tcp
- Sorry nerds, no stats.

- But, it only works on tcpdump files of up to X domains at time. This is because its programmer sucks. Scalability issues.

What about Heyoka?

- New DNS tunnel tool. Not yet publicly available.
- Spoofs source addresses to create asymmetrical DNS tunnel.
- I would **guess** that dnsTTrap will find it. (strictly a guess)
- dnsTTrap is asymmetrical, and it looks at data to a domain, not from hosts.

How does it work cont.

- Its all about tuning
- Able to tune to super low false positive on small networks and single hosts.
- But over-fitting resulted in false-negatives on larger network samples.

- So as a rule, tune it down, but don’t over do it. You may just have to accept some false positives.

Ways to defeat it…

- So, it’s a good idea, but it has a few weaknesses.
  - Don’t use the LLD, use a middle sub-domain. But, this is kinda lame because its an attack on my lame programming ability more than the idea. (iodine, tcp2dns, ozzyman all use LLD).
  - Use tons of domains and make requests multiple times to each. This isnt much of a victory though because your limited DNS-tunnel 3K bandwidth will be cut even further.

This slide used to say something obnoxious

- And has been replaced.

- Slides and source can be found at
- www.meanypants.com
- project email can be sent to
  - themeanypants@gmail.com
- No need to email me about how much my code sucks, I already know.

Thanks!

- So, I have no idea what I’m going to do with this.
- I’m not really interested in patents and the like. Everything I’ve done is public domain, so feel free to work with it.

- Thanks to Hick.org, Skape, Warlord, Rizo, Slurbo and Bill Swearingen for the help and reviews.
- Needs a complete code rewrite, who has that kind of time… ?

Thanks!

- So, I have no idea what I’m going to do with this.
- I’m not really interested in patents and the like. Everything I’ve done is public domain, so feel free to work with it.
- Thanks to Hick.org, Skape, Warlord, Rizo and Bill Swearingen for the help and reviews.
- Needs a complete code rewrite, who has that kind of time… ?

Questions and “Where you can go from here” for the AI bound hacker

- Recommended Reading:
  - Mess around with an AI tool like Weka.
  - Read the internet, wiki actually has really good AI stuff.
  - The best introductory book is “Fuzzy Thinking” by Bart Kosko.
  - NeuroSolutions has the best gui around an ANN. It’s a gui that shows all the innards of an ANN.