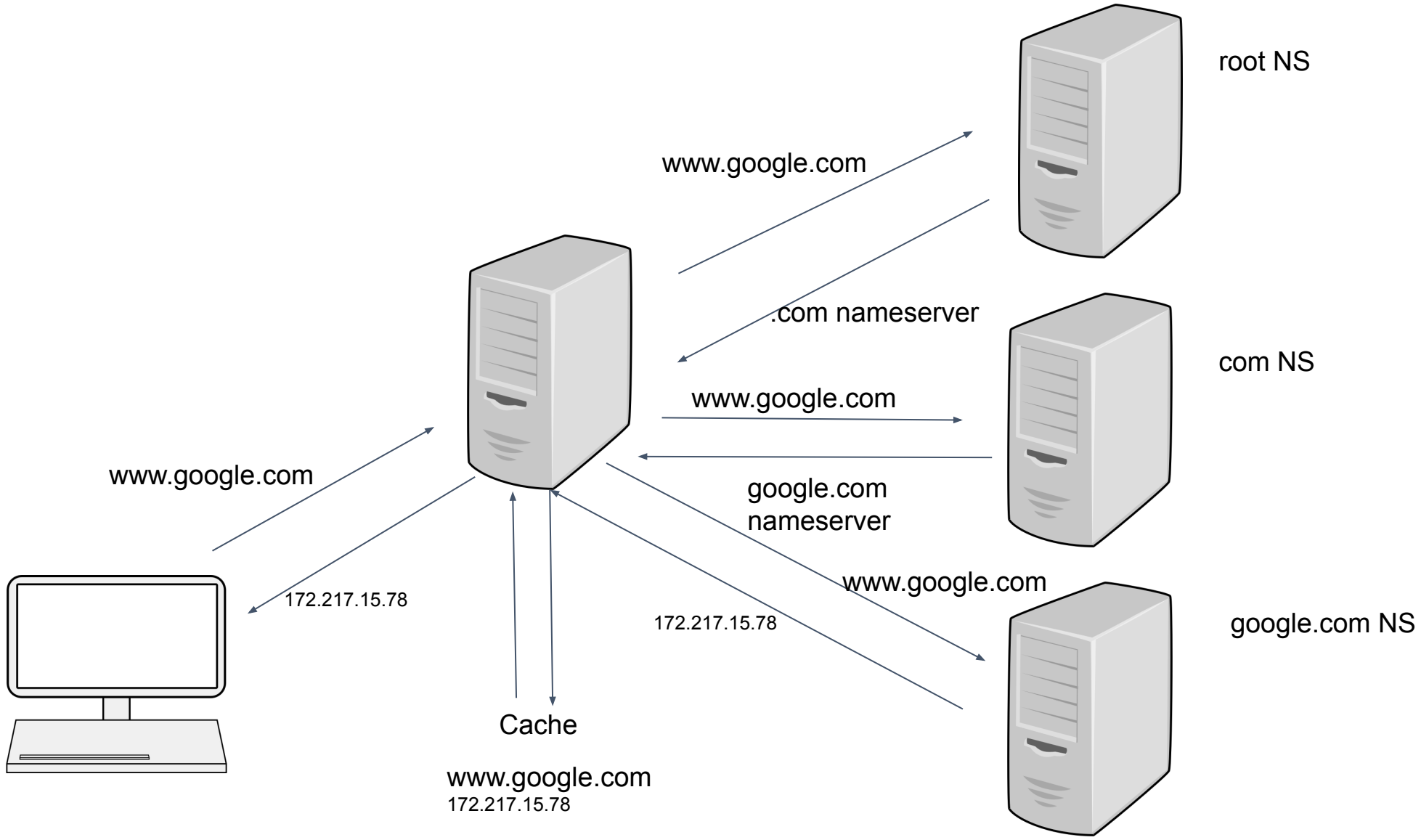
Infoblox

Detecting DNS Tunneling using Behavioral and Content Metadata Features

Prepared by: Darin Johnson, Vadym Tymchenko

Obligatory DNS Recursion Overview



Data: Passive DNS Logs

group_id	Timestamp	qip	qname	qtype	rcode	rrr1
11131	2022-05-01 00:01:59	192.168.4.2	google.com	1	0	[8.8.8.8, 500]
11111	2022-05-01 00:01:59	192.168.4.1	2po3asvtjvfkebjuke4qsvf3ja6agsznrt.12237.2b.dd...	1	0	[35.168.95.233, 0]
54111	2022-05-01 00:01:59	192.168.4.6	ec2-53-24-23-123.west.amazonaws..com	1	0	[53.24.23.123, 500]
11111	2022-05-01 00:02:43	192.168.4.1	2jp99skzob5nzr3o7bjgoemqopvsnvztmfy.12238.2b.dd..	1	0	[205.170.107.30, 300]
11111	2022-05-01 00:02:59	192.168.4.1	jkeewpejfp5rcw8yvcerrcfh4qkc34ckou.12112.2b.dd	1	0	[33.248.144.185, 300]
54111	2022-05-01 00:01:59	192.168.4.3	pmyy.<blahblahblah>.pdrv2.proo...	16	0	["reject rscore=100", 500]
11111	2022-05-01 00:03:01	192.168.4.1	kf1j4m7rsoty56ccreewe4n3u3o4kewmax.12132.2b.dd...	1	0	[35.168.95.233, 0]

DNS Tunneling

Creating a protocol to send/receive data via DNS, other than the actual intended information of DNS RFC's.

Uses

- Early 2000's DEFCON/Blackhat talks
- Command and Control (C2)/Remote Access Tools (RAT)
- Breaking out of "Walled Gardens", i.e. checking your email without paying GOGO-inflight
- Build-your-own RPC (McAfee, zVelo, Spamhaus, e5.sk)

Recent news:

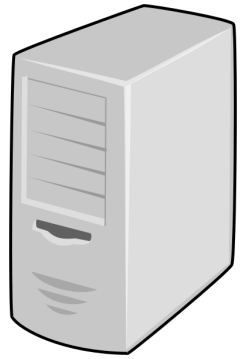
- Saitama
- DNS Anchor

Tools:

- DNSCAT (in METASPLOIT)
- Cobalt Strike
- Weazel (Facebook RAT)
- Iodine

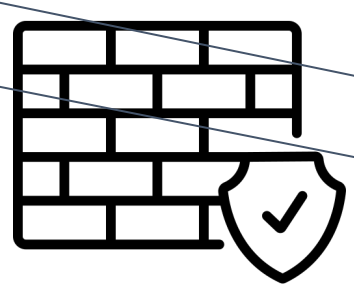
Idea behind tunneling:

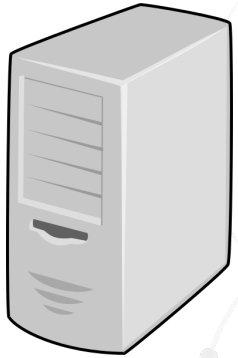
Creating a protocol to send/receive data via DNS, other than the actual intended information of DNS RFC's.



Largely Unmonitored

Receive beacon, Send commands via answer records. Receive responses to commands in qname.





DNS Server

Beacon, receive commands via answer records. Send responses to commands in qname.

Command and Control DNS Server

Blocked by firewall
Break and inspect
…

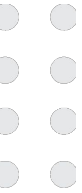
qname: MJSWCY3PNY.mydomain.com
qtype: 16
rcode: 0
answer: ZGlyIGM6XERvY3VtZW50cw==

Just requires a registered domain and control of the DNS server

qname: MJSWCY3PNY.mydomain.com
qtype: 16

Goal: Detect DNS Tunnels

- **Low False Positive Rate**
- **Avoid difficult training set construction**
- **Diversity: Able to detect novel tunnels**



Typical Features - Identifying Domains as Tunnels

- Typical Features
  - Content features:
    - bigrams of qname or prefix of qname (Google ngrams)
    - Compression Ratios of qnames
    - Hidden Markov Models
  - Metadata features:
    - counts
    - entropy of qname $\sum_{c \in word} p_c \log(p_c)$
    - gini of qname $\sum_{c \in word} p_c^2$
    - Length of qname
    - Unique Qname Counts
    - Time Series features (mean times between queries)
  - Aggregated or Smoothed (Averaged) at the Second Level Domain (SLD) and time window.

Metadata Features - Identifying SLDs as Tunnels

We added:

- Cumulative Entropy (Qname, Answers)
- Unique Answer Counts
- Smoothing Higher Order Statistics (Variance)
- Novelty Detection

Useful Enrichments used later but the subject of this talk:

- Popular Domains Inforanks, Umbrella (Not Alexa)
- Name Server Reputation (or # of domains hosted)
- Autonomous System Numbers

Final Features

Summaries at the (Domain, group_id, qip, time window 1 minute)

Name	Description
median_event_entropy	median event entropy
var_event_entropy	variance event entropy
cumulative_prefix_entropy	cumulative prefix entropy
cumulative_answer_entropy	cumulative answer entropy
uniq_qname_ratio	(unique qname count)/(query count)
uniq_answer_ratio	(unique answer count)/(query count)
uniq_qname_answer_ratio	(unique qname count)/(unique answer count)
median_ae_loss	Autoencoder - Will Discuss
var_ae_loss	Will Discuss
median_anomaly_loss	Anomaly Detection - Will Discuss
var_anomaly_loss	Will Discuss

Metadata Experiments

- Calculated Summaries over 1 minute intervals for (group_ip, qip, domain)
- Require minimum of 4 queries per summary.
- Utilized set of blocklist domains as positive class, assume all other domains in negative class.
- Down Sampled negative class (10%)
- Trained a classifier on a training set and reported the results on a test set as normal (checking if any “false positives” were in fact tunnels or RPC’s).
- Tested the trained model on multiple days on data to determine whether the model generalized well over multiple days.
- Verified the classifier could also detect tunnels from pentests which weren’t in the training traffic.

Lazy Dataset construction: Blocklists as a proxy for tunnels

Positive Examples used in training:
abuseat.org, e5.sk,sophosxl.net, spamhaus.org, zvelo.com, mcafee.com, … 50 more

Assumption: Everything else is not a tunnel!

Class Imbalance:

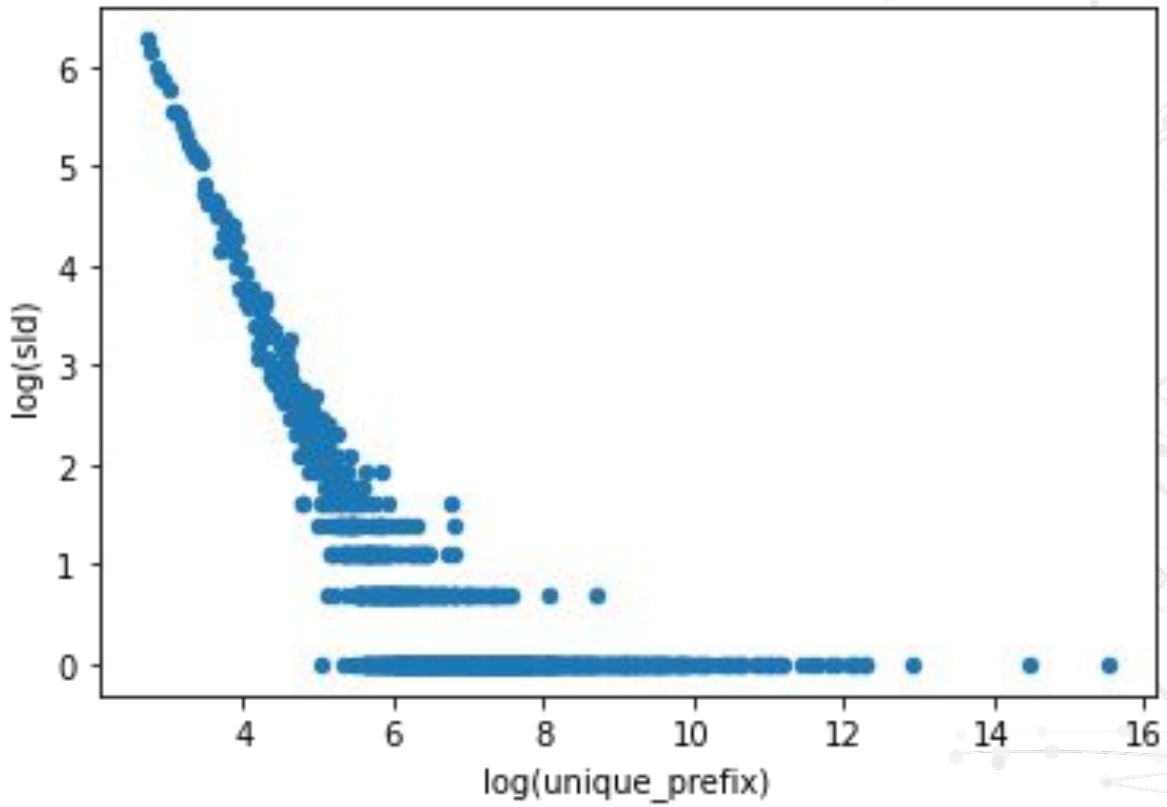
tunnel	count	percent
True	44735	1.3%
False	3275000	98.7%

Content Features: ae_loss and anomaly_score

- Our approaches:
  - CountVectorizer(analyzer='char', ngram=[1,4]) + Anomaly Detector (pca, iso-for/ocsvm) Pipeline
  - Character-level Autoencoder
- Novelty Detection

Domain prefixes degression

- A prefix is the front part of an FQDN before the “effective second level domain”. I.e prefix(www.mail.mirc.co.uk)=www.mail
- Prefixes prepended to more than 15 slds, account for on 20% of DNS traffic (<6000 such prefixes)
  - www
  - smtp
  - ftp
  - default._bimi
  - _dmarc
  - _acme-challenge
- There are 13 million prefixes observed once per day but they aren’t anomalous.
  - ec2-123-23-2-1.west
  - ip-21-32-4-2
  - …
  - and trackers



anomaly_score: N-Gram vectorization

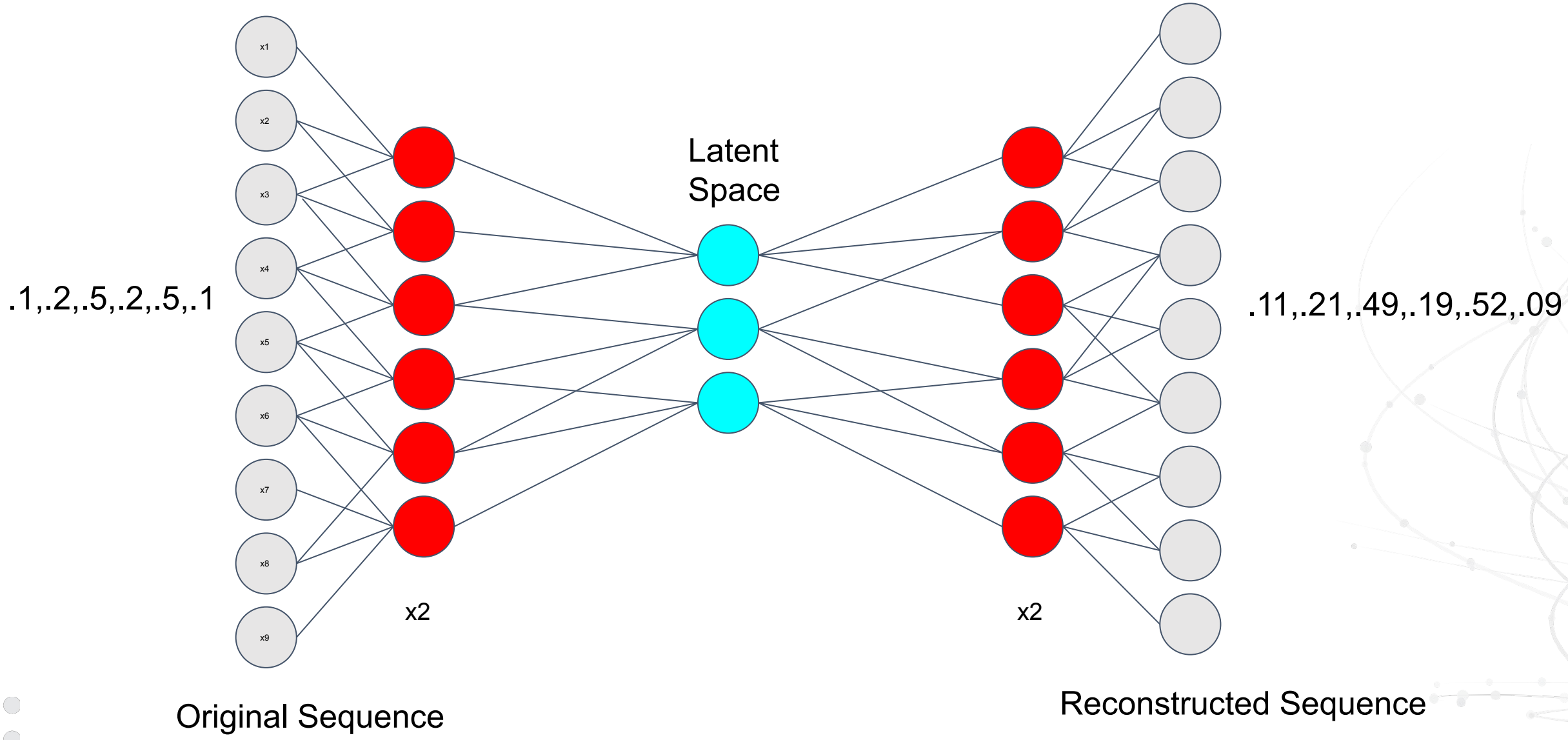
ec2-123-23-2-1.west → [ec2, c2-, 2-1, 123, 23-, 3-2, -23, 23-,3-2, -2-,-1., .1.w, .we, wes, est]

→ [1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1,1, 0….] →

Favorite One Class Classifier

- ocsvm
- isolation forest
- …

Convolutional Autoencoders



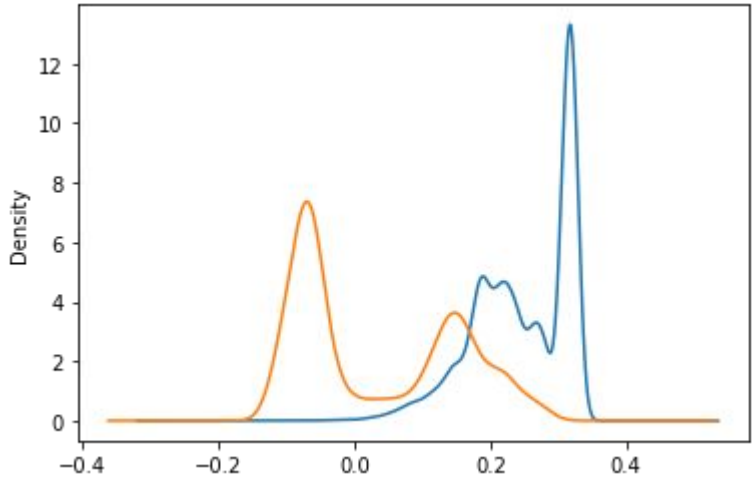
Reconstructions at end of Epoch 0

Reconstruction Loss (NLL)	Original	Reconstruction
5.517239570617676	hello world	hh==ob¯ox=° VVVVVVVVVVVVVVVVV
5.5572590827941895	76caec8c171c2a3d5063ab1a7ad3f51e5 be727a85dc50f8bda313c6d009c1b	76ophobo171oip□°006□pb1p7p°□ p01h0bh7i7pb0°o00pbb°p
5.560171604156494	d.tx.17c3400a7.4c750fba.dns	°×tb×17o□Í00p7×Ío700pbp×° noVVVVVVVVVVV
5.560108184814453	www1	¯¯¯1

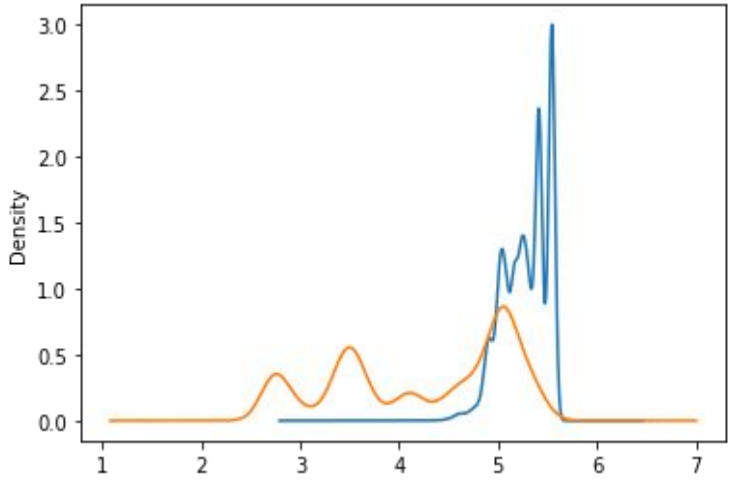
Reconstructions at end of Epoch 50

Reconstruction Loss (NLL)	Original	Reconstruction
5.494847297668457	hello world	hello_worl
5.580347537994385	76caec8c171c2a3d5063ab1a7ad3f51e5 be727a85dc50f8bda313c6d009c1b	76caec8c171c2a3d5063ab1a7ad3f51e5 be727a85dc50f8bda313c6d009c1b
5.545787811279297	d.tx.17c3400a7.4c750fba.dns	d.tx.17c3400a7.4c750fba.dn
5.5392656326293945	www1	www1

Content features to distinguish between standard and tunneling traffic.

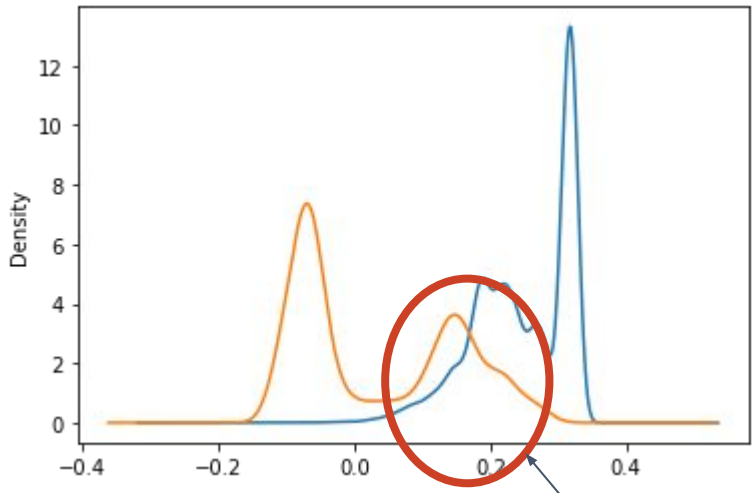


loss2: ngram anomaly detection
Standard traffic (blue) vs tunneling traffic (orange)

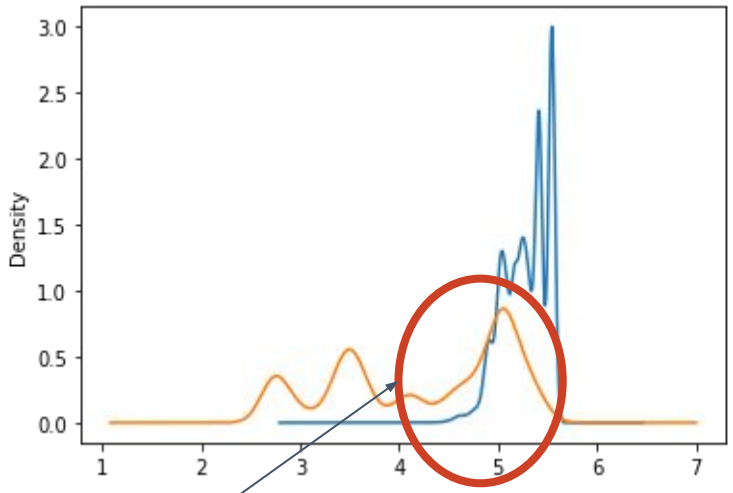


loss1: Reconstruction loss of neural net
Standard traffic (blue) vs tunneling traffic (orange)

Content features to distinguish between standard and tunneling traffic.



ngram anomaly detection
Standard traffic (blue) vs
tunnelling traffic (orange)



Reconstruction loss of neural net
Standard traffic (blue) vs
tunnelling traffic (orange)

What is this interestion?

Intersection: Some of the block lists just use IPs or domains, this is similar to ISPs and/or CDNs

Blocklist	CDN	Internet provider
example.com.domaincheck.blocklist.com	example.com.cdn.cdn-domain.com	
54.23.23.2.ipcheck.blocklist.com		54-23-23-2.dsl.provider.com
5057000fb26a0e956eb52caef63c4b8a.hashcheck.blocklist.com		
example.com.<guid>.domaincheck.blocklist.com		
54.23.23.2.<guid>.ipcheck.blocklist.com		

More prudent list of blocklist domains: mail-abuse.com, barracudabrts.com, sophosxl.net, e5.sk, surriel.com, spamhaus.net, trendmicro.com, mcafee.com, cnr.io, nessus.org, sophosxl.com, zvelo.com

A Valid Question: Why didn't I use the encoder features or output of the NGram vectorization?

Both would use string based features to separate the positive and negative classes. This would diminish the ability to find new tunnels the differed from the tunnels used in training.
Likely we'd learn to separate on strings like:

- xbl
- phish2
- possibly parts of guids

Results

Held out test set (class imbalance: 5.7%)

Accuracy:	99.7%
Precision (Classified and True Positive/Classified Positive):	99.3%
Recall (Classified and True Positive/True Positive):	96.1%

Held out days (class imbalance: < 1.0%), (ADJ - scores adjusted for held out domains)

	Day 1 (ADJ)	Day 2 (ADJ)	Day 3 (ADJ)	Day 10 (ADJ)
Accuracy:	99.8% (99.7%)	99.8% (99.7%)	99.8% (99.6%)	99.8% (99.6%)
Precision (Classified and True Positive/Classified Positive):	92.7% (93.7%)	92.9% (93.6%)	91.0% (92.3%)	88.6% (91.7%)
Recall (Classified and True Positive/True Positive):	96.3% (87.4%)	97.0% (89.0%)	96.6% (79.3%)	93.5% (65.1%)

The False Positives

In 4 days: 208 unique domains total (~120 second level domains per day, ~100 in intersection) - we see ~3.8 million second level domains a day.
Categories:

- 5 Infoblox tunneling domain use for sales demos (Not used in training)
- 4 “interesting” domains (No longer resolving, similar pattern)
- 3 configuration issues (related to above domains)
- 13 blocklist domains (Not used in training)
- 32 Social Media/Ad Trackers/Metrics
- 151 CDN/Hosting providers (much less than previous tooling, in variety and volume)
  - 3 that I had to look-up (short odd prefixes and a lot of ASNs - but on permit lists)

With standard allowlists (TopN, Blocklist Filter and Nameserver Reputation):

- 5 Infoblox tunneling domain use for sales demos (Not used in training)
- 4 “interesting” domains (No longer resolving, similar pattern)
- 5 Trackers

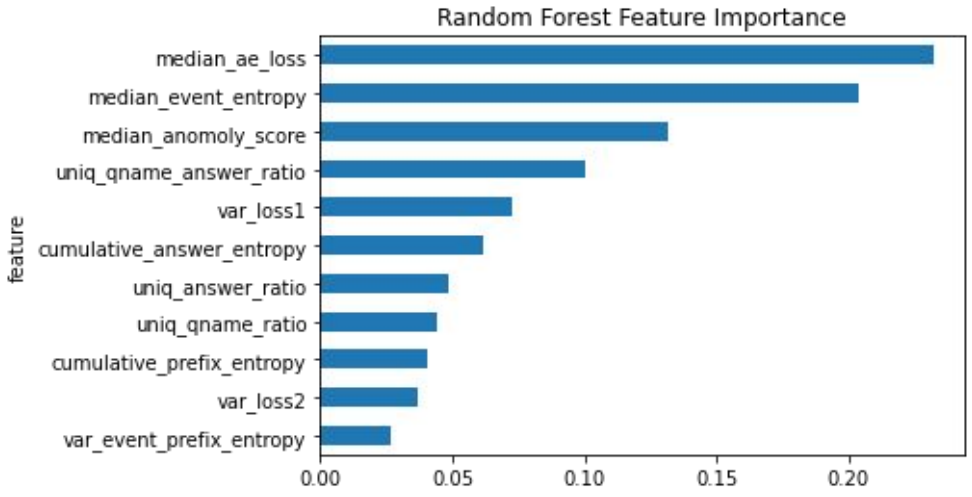
Recall: Does this really work?

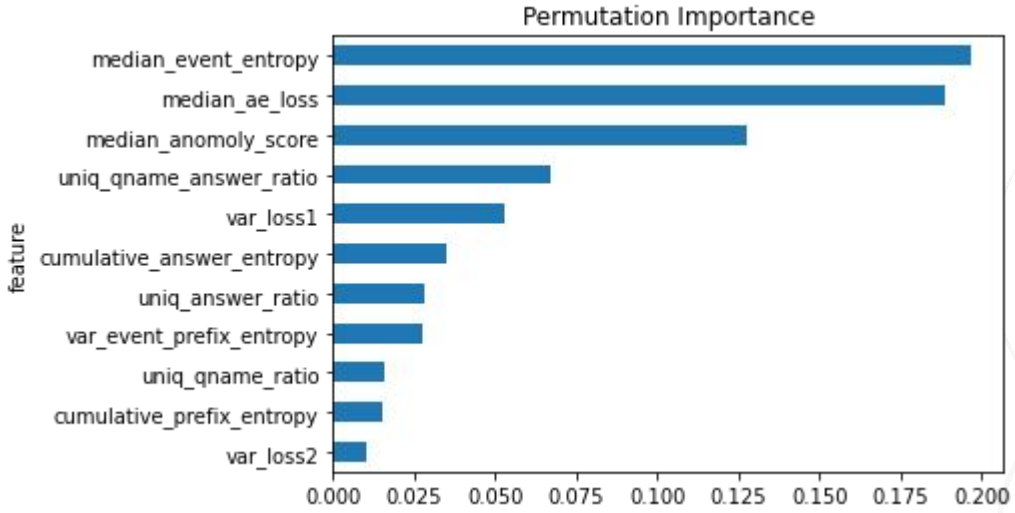
Replayed data from two pen tests (different software used) our customers performed. Correctly labeled 91% of the summaries as tunnels.

Assuming uniform time distribution of pos/negative labels, 99.2% chance of discovery with 2 minutes.

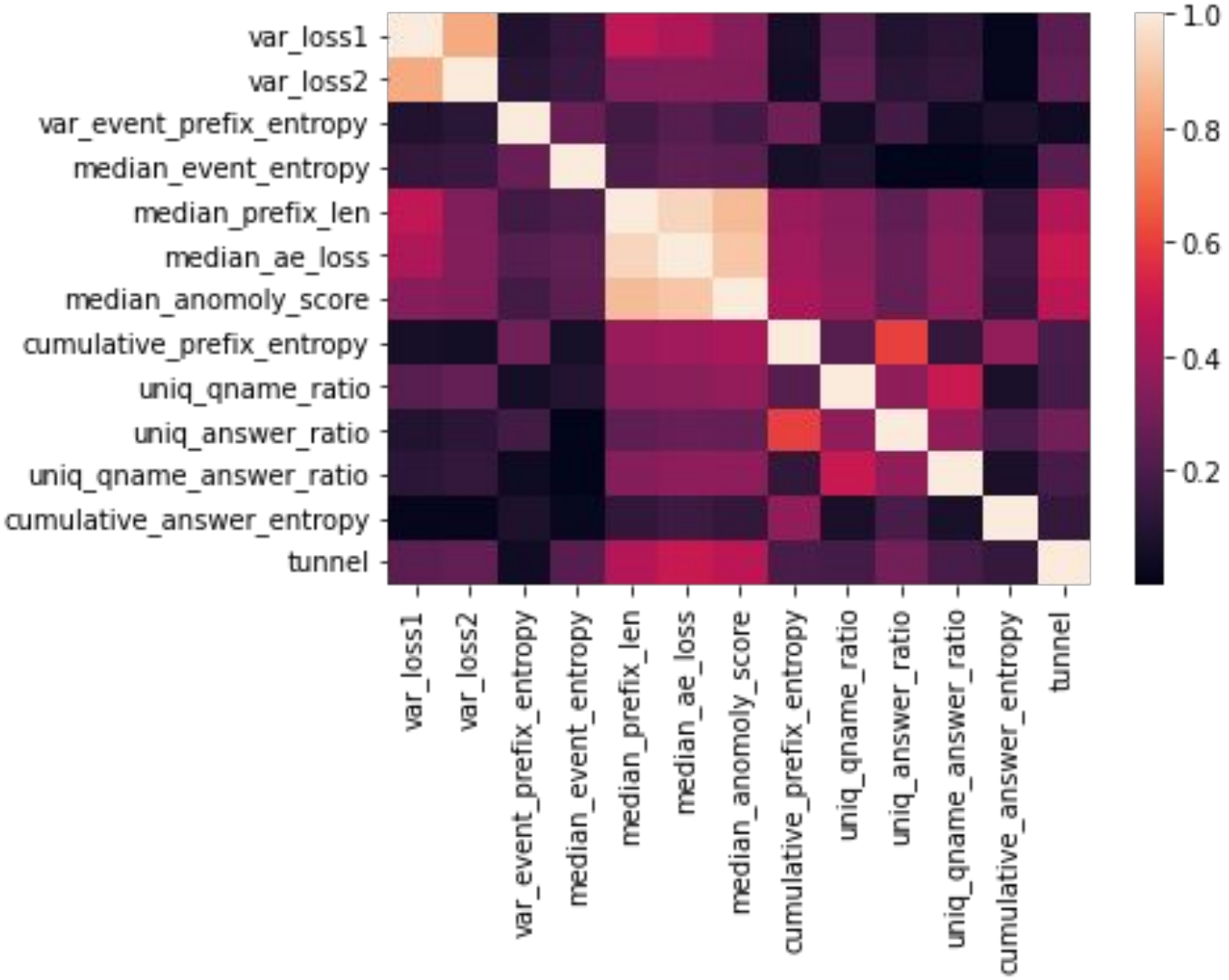
NB: One minute summaries won’t detect beacons.

Feature Importances





Feature Importances



Feature Importance: Answers Matter Some

Removing Answer Information: uniq_qname_answer_ratio, uniq_answer_ratio, and cumulative_answer_entropy

Held out test set (class imbalance: 5.7%)

Accuracy:	99.3% (down 0.3%)
Precision (Classified and True Pos/Classified Pos):	98.8% (down 0.5%)
Recall (Classified and True Pos/True Pos):	92.2% (down 3.9%)

Feature Importance: Prefix structure matters alot

Removing median_ae_loss, median_anomaly_score, adding median_prefix_len

Held out test set (class imbalance:5.7%)

Accuracy:	99.1% (down 0.5%)
Precision (Classified and True Pos/Classified Pos):	98.5% (down 1.2%)
Recall (Classified and True Pos/True Pos):	90.1% (down 6.0%)

Conclusions:

- You can detect novel tunnels in DNS traffic
- Block lists seem to make a decent proxy for tunnels
- Deep learning can help derive useful qname features
- Useful information in in the answers
- Some useful features not in the logs
  - TopN Domains (Inforanks, Umbrella)
  - Unique ASN count
  - Nameserver Reputation
  - Expected: Logs don’t convey all information about a domain

Next Steps:

- **Time resolutions: Detect slower C2, expect more false positives.**
- **Clustering anomalous data: Using the latent encoder space, automate identification of similar blocklists, tunnels, or other interesting domains.**
- **Do we have the right NN architecture? (I don’t think so, but it works for now - LTR)**
  - **Adding Qtype and Answers to the reconstruction?**

References:

- Dahan, Assaf. “Dropping Anchor: From a TrickBot Infection to the Discovery of the Anchor Malware.” Cybereason, 11 December 2019, https://www.cybereason.com/blog/research/dropping-anchor-from-a-trickbot-infection-to-the-discovery-of-the-anchor-malware#conclusion. Accessed 4 August 2022.
- “iagox86/dnscat2.” GitHub, https://github.com/iagox86/dnscat2. Accessed 3 August 2022.
- “RFC 5782 - DNS Blacklists and Whitelists.” IETF Tools, https://datatracker.ietf.org/doc/html/rfc5782. Accessed 3 August 2022.
- Schouten, Tom. “yarrick/iodine: Official git repo for iodine dns tunnel.” GitHub, https://github.com/yarrick/iodine. Accessed 3 August 2022.
- Stockley, Mark. “How the Saitama backdoor uses DNS tunnelling.” Malwarebytes Labs, 25 May 2022, https://blog.malwarebytes.com/threat-intelligence/2022/05/how-the-saitama-backdoor-uses-dns-tunnelling/. Accessed 4 August 2022.
- Yu, Bin, et al. “Behavior Analysis based DNS Tunneling Detection and Classification with Big Data Technologies.”


Thank You
Asante
Shukria
Dhanyavadagalu
Vinaka
köszönöm
Kiitos
Maake
Manana
Dankon
Matondo
감사합니다
Kam Sah Hammida
شكرا
Dank Je
Mauruuru
Biyan
Terima Kasih
Taiku
Tack
Blagodaram
Dankscheen
спасибо
Chokrane
Diolch i Chi
Ngiyabonga
Dziekuje
Grazie
Arigato
Juspaxar
Mochchakkeram
Gracias
Tingki
நன்றி
Bedankt
धन्यवाद
cảm ơn bạn
Dakujem
Gràzas
Gratias Tibi
Ua Tsaug Rau Koj
Paldies
Khun Khap
Obrigado
Děkuji
Nirringrazzjak
Kia Ora
ありがとう
Suksama
Hvala
Kop
Djiere Dieuf
Matur Nuwun
Rahmat
谢
Welalin
Di Ou Mèsi
Eskerrik Asko
Misaotra
Danke
Go Raibh Maith Agat
Merci
ขอขอบคุณคุณ
Salamat
Najis Tuke

DNS: Not just A record or more than 1 (q)type

- Many more: A (ipv4), AAAA(ipv6), TXT, MX (Mail), SRV, NULL, NS, CNAME(canonical name)
  See the wikipedia page for a mostly up to date list

DNS: Not just A record or more than 1 (q)type

- A **TXT record** (or **text record**) is a DNS resource record that associates arbitrary text with a host or other name. Generally this is “human readable” information about a server, network, or other information system.
- **Uses**:
  - ACME Protocol, used by Let’s Encrypt (others?) to verify Domain Ownership prior to distributing SSL Certificates
  - SPF, DMARC, DKIM, BIMI - Ensures proper identification of mail servers, spam prevention and logo information
  - Site Verification (Google, Adobe, etc)
  - Really whatever you want:
    - McAfee, Zvelo, Team Cymru send file hashes and report if malware
    - Spamhaus, Spamcop receive IP Addresses/Domains and report spam information
    - e5.sk is a content control system
    - Cryptocurrency
    - Build-your-own RPC system
- Many more: MX (Mail), SRV, NULL, NS, CNAME(canonical name), A (ip), AAAA(ipv4)
  [See the wikipedia page for a mostly up to date list]

Data: Passive DNS Logs

- **~15 Billion events, DNS Responses below the resolver.**
- **Relevant Fields:**

Name	Type	Description
group_id	Integer	unique group_identifier
qip	String	IP address or the client originating the request
qname	String	Fully Qualified Domain Name or the query
rrr1	List[answer: String, ttl: Integer]	List of answers returned for query
timestamp	Integer	Timestamp
qtype	Integer	Type of query (IP, MX, TXT, etc)
rcode	Integer	Success or Failure (and reason) of query

N-Gram vectorization

3-grams:
ec2-123-23-2-1.west becomes {‘ec2’: 1, ‘c2-’: 1, ‘2-1’:2, ‘123’: 1, ‘23-’: 2, ‘3-2’: 1, ‘-2-’:1 }.

Doing this for all prefixes creates a sparse vector space. We can now use standard one class classifiers like isolation forests or ocsvm to do novelty detection.

Idea: Find things different from normal traffic.

Lazy Dataset construction: Blocklists as a proxy for tunnels

Positive Examples used in training:
abuseat.org, ahbl.org, atlbl.net, baracudacentral.org, blocklist.de, cymru.com, dnswl.org, drand.net, dronebl.org, e5.sk, fabel.dk, gbudb.net, hostkarma.com, inps.de, ipquery.org, junkemailfilter.com, lookout.com, manitu.net, nessus.org, njabl.org, orbitrbl.com, proofpoint.com, proxybl.org, rfc-ignorant.org, senderbase.org, sophosxl.net, sorbs.net, spamcannibal.org, spamcop.net, spameatingmonkey.com, spamhaus.org, spamrats.com, surbl.org, surriel.com, tiopan.com, trendmicro.com, uceprotect.net, unsubscore.com, v4bl.org, wpbl.info, zvelo.com, skydns.ru, mail-abuse.com, surfsrs.com, spamhaus.net, mcafee.com, rspamd.com, sare.net, sendgrid.net

Assumption: Everything else is not a tunnel!

Class Imbalance:

tunnel	count	percent
True	44735	1.3%
False	3275000	98.7%