中国科学院 信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

Leopard: Understanding the Threat of Blockchain Domain Name Based Malware

Zhangrong Huang[1,2], Ji Huang[1,2], and Tianning Zang[2]

1.School of Cyber Security, UCAS

2.Institute of Information Engineering, CAS

PAM 2020

verizon media COMCAST Akamai

Existing Techniques Used by Malware

- IP Flux

  IP Flux is a technique which enables malware change IP addresses of their C&C servers.

evil.domain.com → 1.2.3.4
evil.domain.com → 2.3.4.5
evil.domain.com → 3.4.5.6
evil.domain.com → 4.5.6.7

- Domain Flux (Domain Generation Algorithm)

  It is another way for malware to evade detection by generating pseudorandom domains or dictionary-based domains of C&C servers.

sdfgsodmsdoj.com 192.168.1.10
sdfijozccbsnqs.com
qwewqpoyuca.com

evil3.ccserver.com 172.16.10.5
evil4.ccserver.com
evil5.ccserver.com

New Threat: Blockchain Domain Name Based Malware

- Blockchain domain based name malware (BDN-based malware) is a new type of malware which leverages Blockchain DNS (BDNS).
- Some authors of malware offered an updated variant of malware that included blockchain domains support.
- More than 140K domains registered in both Namecoin and Emercoin.
- Pioneers of Blockchain DNS.

Advertisement Translated Text:

AZORult V2

[+] added .bit domains support
[+] added CC stealing feature (for Chrome-based browsers)
[+] added passwords grabbing from FTP-client WinSCP
[+] added passwords grabbing from Outlook (up to the last version)
[+] fixed passwords grabbing from Firefox and Thunderbird

(Figure is from FireEye report)

namecoin Blockchain OpenNIC Project

[1] FireEyE report: https://www.fireeye.com/blog/threat-research/2018/04/cryptocurrencies-cyber-crime-blockchain-infrastructure-use.html

PAM 2020 verizon media COMCAST Akamai

Related Works

- Patsakis C. et al. analyzed related security issues of introducing blockchain-based DNS and offered some advice to mitigate corresponding threats.

- Pleiades, FANCI, Error-Sensor, and BotMiner: They are prior works of detecting malware (botnet) based on error information, DNS traffic or HTTPS traffic.

- Drawback: No suitable solutions to detecting malicious blockchain domains, due to the special mechanism of BDNS

Our Contributions

- Leopard: The first prototype of the automatic detection of malicious blockchain domains (BDNs).

- Great performance: System reaches an AUC of 0.9980 on the real-world datasets and it has an ability to discover 286 unknown malicious BDNs.

- Two datasets: The set of malicious BDNs and the list of DNS servers providing BDNs resolution service.

Outline

1. Background

2. Automatic Detection

3. Evaluation

4. Limitations

5. Conclusion

Outline

1. **Background**

2. Automatic Detection

3. Evaluation

4. Limitations

5. Conclusion

Blockchain Domains

- Blockchain domains have special TLDs that different from generic TLDs and country-code TLDs.

Organizations	TLDs	DNS Servers
Namecoin	.bit	-
Emercoin	.coin .emc .lib .bazar	seed1.emercoin.com seed1.emercoin.com

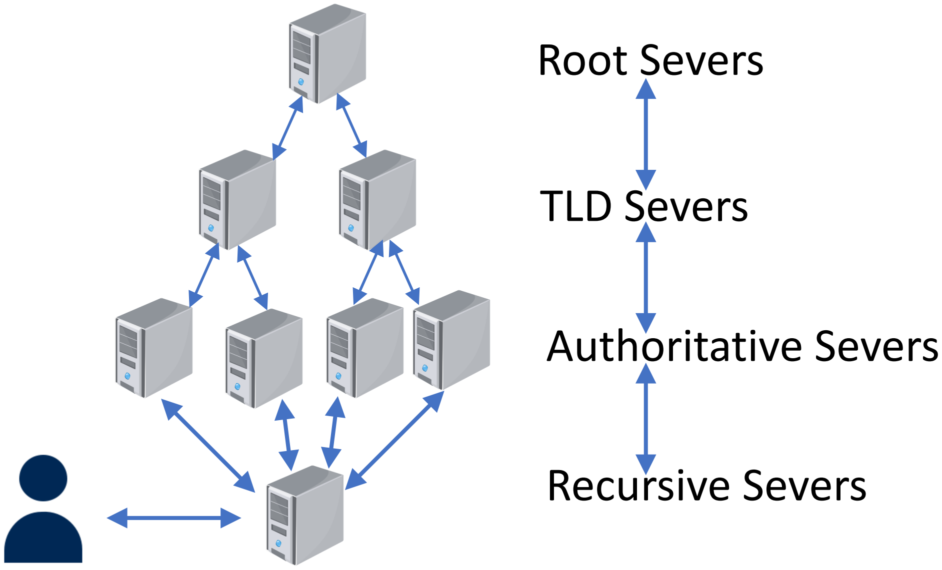
- Blockchain domains are of inherent properties.
  - Anonymity
  - Censorship-resistance

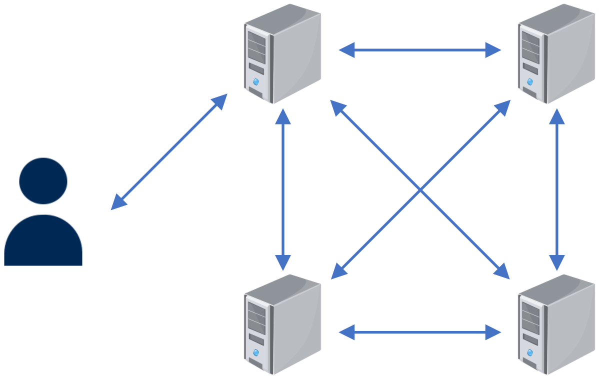
Name: dns:alibaba.bazar
Value History:
BM-NC3ZPQGeD6DV3cZQnTWhh3DviQveWEvp
BM-NC3ZPQGeD6DV3cZQnTWhh3DviQveWEvp
BM-NC3ZPQGeD6DV3cZQnTWhh3DviQveWEvp
BM-NC3ZPQGeD6DV3cZQnTWhh3DviQveWEvp

Owner: EPQTyZvVAJ39oDmNhR17zut6sETnFY7n63
Valid until: 08.01.2062

[1] Block 103341 :https://explorer.emercoin.com/block/103341

Blockchain DNS (Architecture)



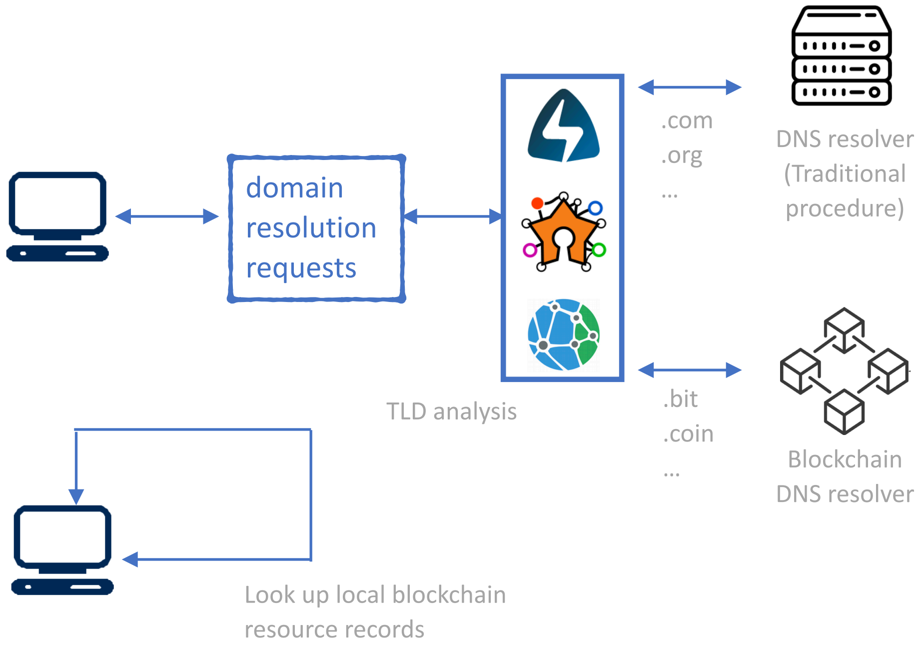


Users can issue a BDN query to any server which has blockchain domain resource records.

Blockchain DNS (Workflow)

- Third-party BDNS
  - Leverage proxy or browser plugins to forward DNS requests to third-party BDNS.

- Local BDNS
  - If users download chains in advance, the requests can be resolved locally.



Outline

1. Background

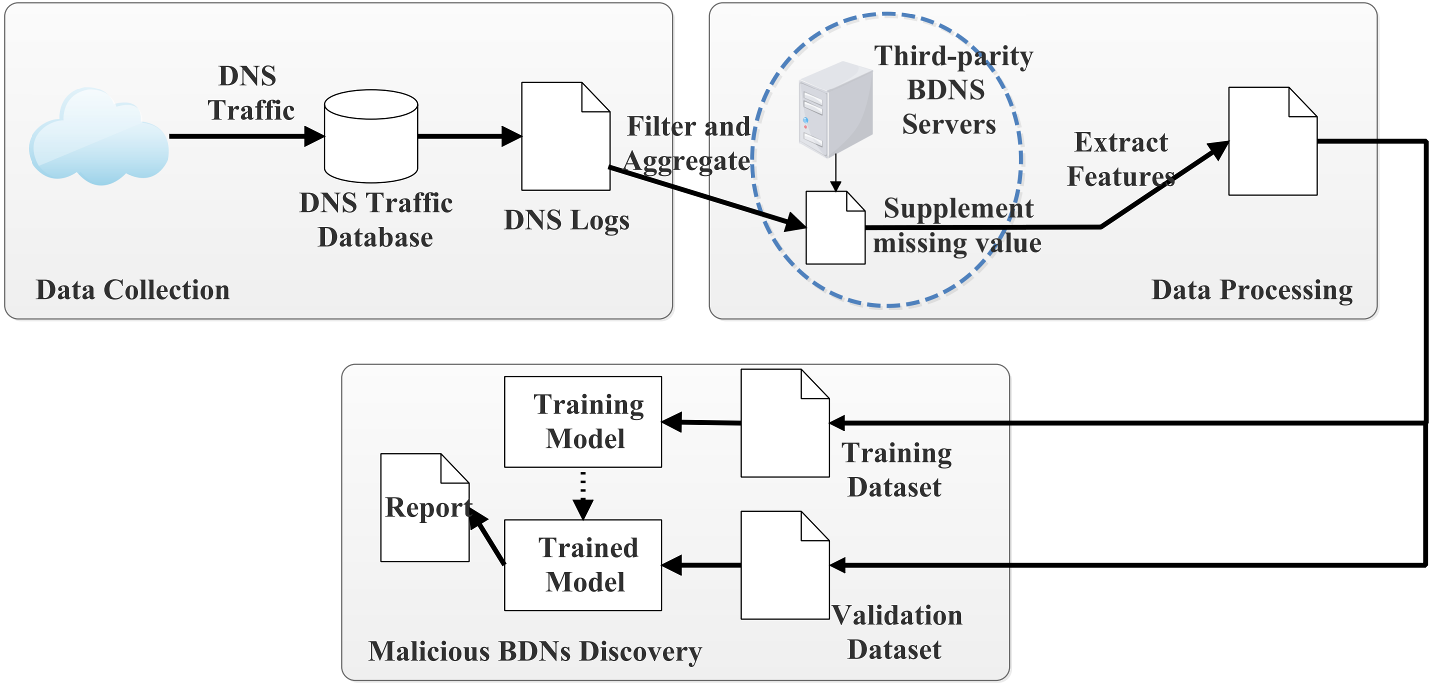
2. **Automatic Detection**

3. Evaluation

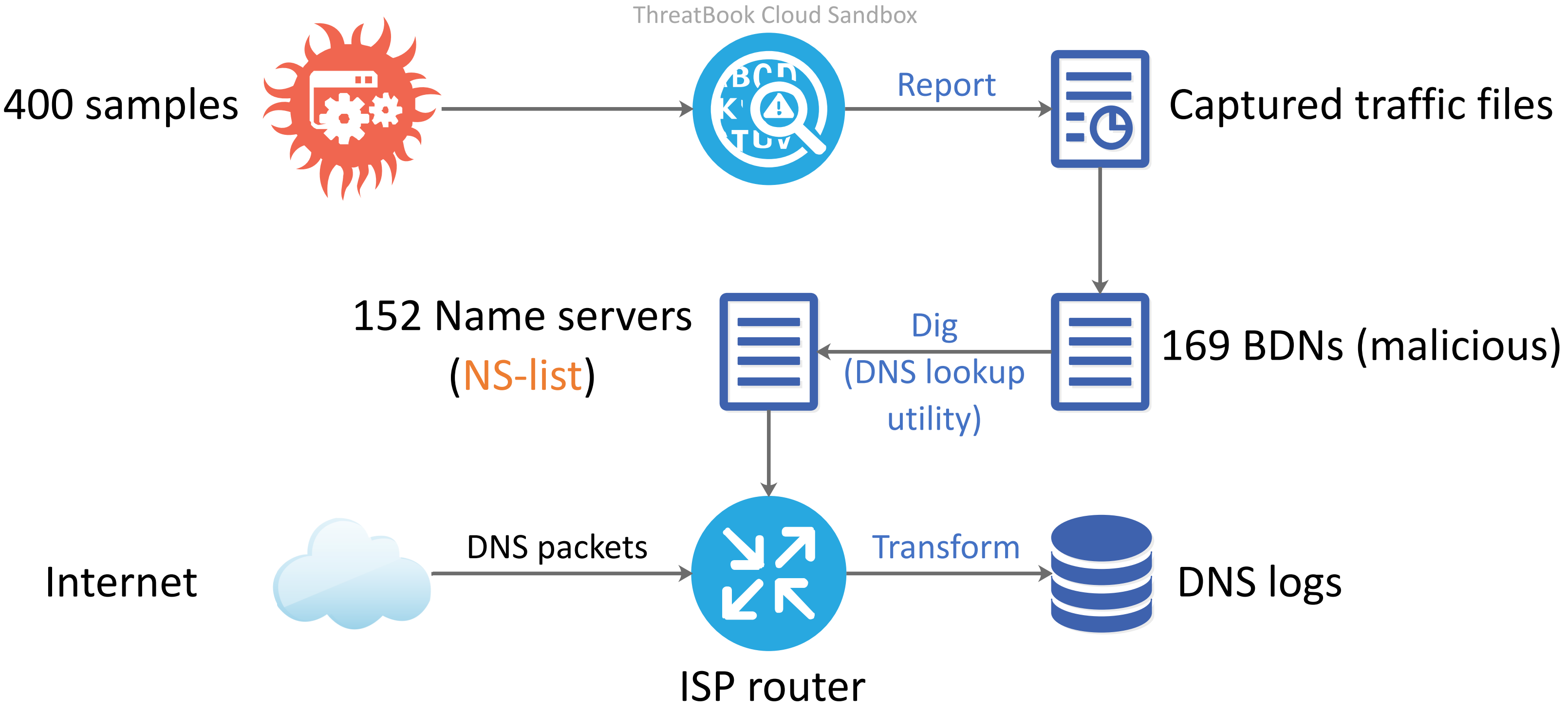
4. Limitations

5. Conclusion

Overview of Leopard



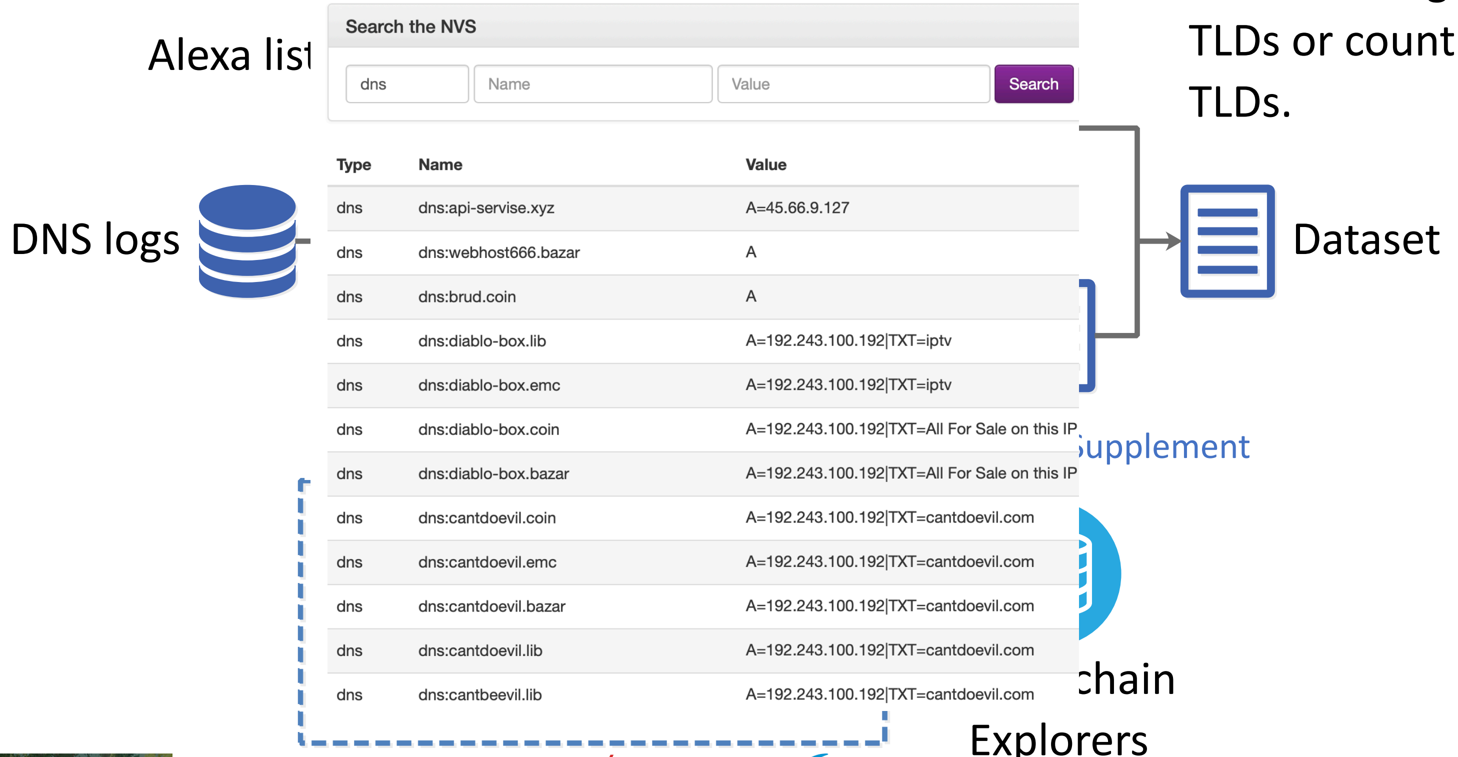
Module (Data Collection)



PAM 2020

verizon media COMCAST Akamai

Module (Data Processing)



Search the NVS

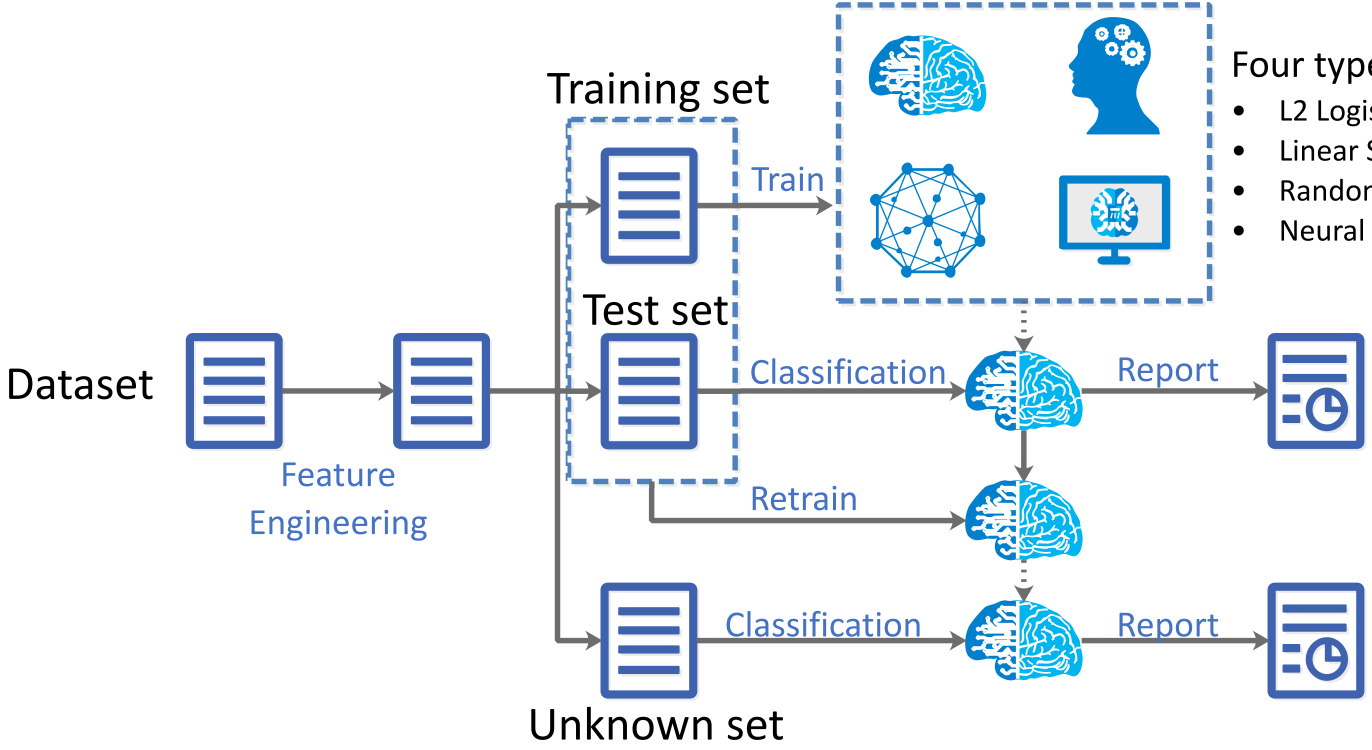
dns | Name | Value | Search

Type	Name	Value
dns	dns:api-servise.xyz	A=45.66.9.127
dns	dns:webhost666.bazar	A
dns	dns:brud.coin	A
dns	dns:diablo-box.lib	A=192.243.100.192|TXT=iptv
dns	dns:diablo-box.emc	A=192.243.100.192|TXT=iptv
dns	dns:diablo-box.coin	A=192.243.100.192|TXT=All For Sale on this IP
dns	dns:diablo-box.bazar	A=192.243.100.192|TXT=All For Sale on this IP
dns	dns:cantdoevil.coin	A=192.243.100.192|TXT=cantdoevil.com
dns	dns:cantdoevil.emc	A=192.243.100.192|TXT=cantdoevil.com
dns	dns:cantdoevil.bazar	A=192.243.100.192|TXT=cantdoevil.com
dns	dns:cantdoevil.lib	A=192.243.100.192|TXT=cantdoevil.com
dns	dns:cantbeevil.lib	A=192.243.100.192|TXT=cantdoevil.com

verizon media COMCAST Akamai

ODNs stands for ordinary domain names with generic TLDs or country-code TLDs.

Module (Malicious BDNs Discovery)



Four types of algorithm:

- L2 Logistic Regression
- Linear Support Vector Machine
- Random Forest
- Neural Network

Outline

1. Background
2. Automatic Detection
3. **Evaluation**
4. Limitations
5. Conclusion

Goals of The System

- Q1: Is the system able to distinguish malicious BDNs in real-world network traffic?

- Q2: Does the system have an ability to detect unknown BDNs (have not been discovered by a vendor like VirusTotal)?

Summary of Datasets

- We collected nine-day traffic (about 59GB raw data) and observed a total of 13,035 IPs.
- Aggregation format:

$$(\text{domain\_name}, \text{request\_IP}) : \text{src\_list}, \text{rdata\_set}$$

$$\text{src\_list} = [(\text{IP}_1, \text{port}_1, \text{time}_1), (\text{IP}_2, \text{port}_2, \text{time}_2), \ldots]$$

$$\text{rdata\_set} = \{(\text{record}_1, \text{ttl}_1), (\text{record}_2, \text{ttl}_2), \ldots\}$$

- Aggregated data were divided into three sets. $D_{unknown}$ only has the records of unknown BDNs.

Table 2. The summary of the daily datasets.

Dataset	# Packets	# Remaining Packets	# Blockchain Domains Packets	# Aggregated Records
D_1	38,258,120	10,431,757	215,095	104,132
D_2	32,269,248	9,235,243	818,722	191,564
D_3	29,418,020	8,486,445	413,467	139,957
D_4	33,177,324	8,938,011	398,898	136,488
D_5	33,195,292	10,477,746	390,216	102,825
D_6	26,940,188	7,770,275	383,729	132,534
D_7	25,767,291	6,010,492	388,978	118,139
D_8	25,370,998	6,227,078	390,026	124,657
D_9	30,977,692	6,582,441	316,279	134,590

Table 3. The datasets for training and testing.

Dataset	# Benign Records	# Malicious Records	# Aggregated Records
D_{train_val}	329,850	709	330,559
D_{test}	147,879	160	148,039
$D_{unknown}$	-	-	403

P M 2020

verizon media COMCAST Akamai

Feature Engineering

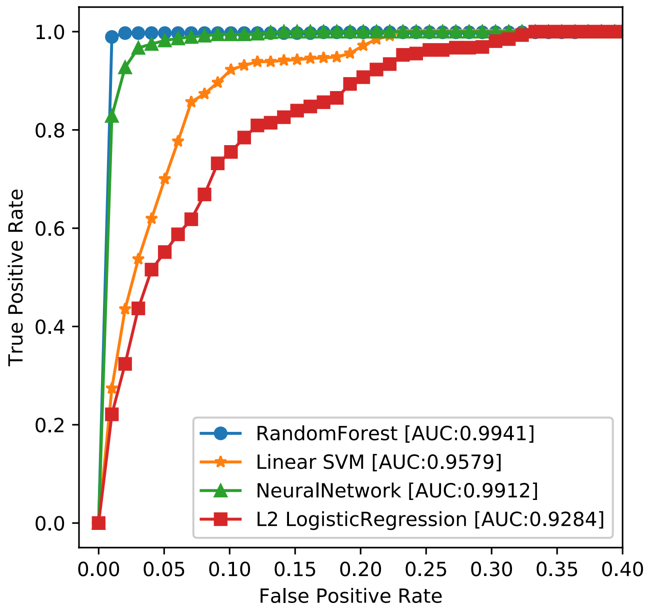
- Three categories of features.
  - Time Sequence feature set
  - Source IP feature set
  - Resource Records feature set

Table 4. Feature Selection

Category	Feature	Feature domain	Novelty
Time Sequence	TimeDiffMin (f1)	Real	New
	TimeDiffMax (f2)	Real	New
	TimeDiffMedian (f3)	Real	New
	TimeDiffStd (f4)	Real	New
	PktNumPerMinMin (f5)	Real	New
	PktNumPerMinMax (f6)	Real	New
	PktNumPerMinMedian (f7)	Real	New
	PktNumPerMinStd (f8)	Real	New
Source IP	SrcIPNum (f9)	Integer	[29]
	ASNNum (f10)	Integer	[23]
	CountryNum (f11)	Integer	[29]
Resource Records	ARecordNum (f12)	Integer	New
	NSRecordNum (f13)	Integer	New
	TTLMin (f14)	Integer	New
	TTLMax (f15)	Integer	New
	TTLMedian(f16)	Integer	New
	TTLStd(f17)	Real	[29]

Cross-Validation on Training Set

- The metric used to evaluate the performance of classifiers is AUC_ROC (the area under the receiver operating characteristic curve).
- The random forest classifier outperforms the other classifiers and reaches an AUC of 0.9941.
- Linear models are not suitable to solve this quite difficult problem.



verizon media COMCAST Akamai

Feature Analysis (1)

- We assessed the importance of each feature through the mean decrease impurity which is a measure of the random forest algorithm to select features.

Table 5. MDIs of the features.

Rank	Feature	Score	Rank	Feature	Score	Rank	Feature	Score
1	f16	0.23220529	7	f13	0.02745297	13	f7	0.01823298
2	f15	0.21952513	8	f9	0.02673914	14	f1	0.01623537
3	f14	0.21214118	9	f8	0.02664864	15	f3	0.01490099
4	f12	0.05541738	10	f11	0.02521994	16	f10	0.01249088
5	f17	0.03356060	11	f6	0.02384570	17	f5	0.00369699
6	f2	0.02831889	12	f4	0.02336793			

Feature Analysis (2)

- Also, the different combinations of feature sets were assessed by training the same classifier with different features.

Table 6. AUCs of the classifiers using the different combinations of the feature sets.

Combinations	AUC
All Features	0.9945
Time Sequence	0.8850
Source IP	0.7920
Resource Records	0.9935
Resource Records + Source IP	0.9944
Resource Records + Time Sequence	0.9944
Time Sequence + Source IP	0.9944

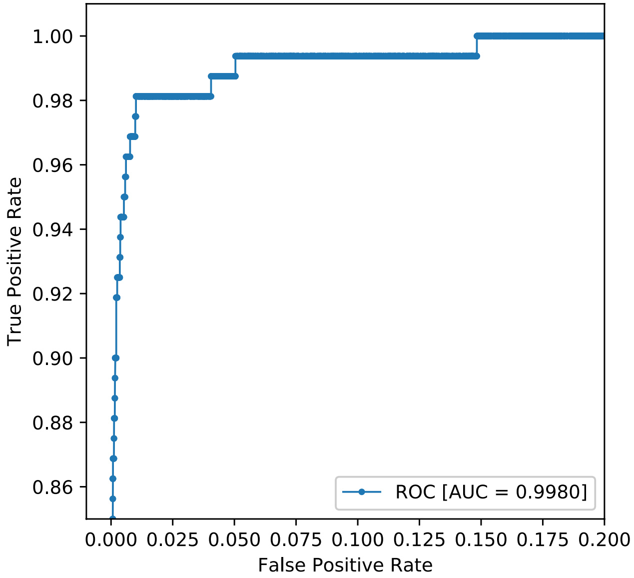
Evaluation on D_{test}

- Leopard achieves an AUC of 0.9980.

- When the detection rate reaches 0.98125, the false positive rate is only 0.1010.

- Q1: Is the system able to distinguish malicious BDNs in real-world network traffic?

  Answer: Leopard can accurately detect malicious BDNs



Evaluation on $D_{unknown}$

- Leopard reported 309 malicious records out of 403 and the reported records included 286 unique BDNs and 23 server IPs.
- Rules to verify the result:
  - Any of the historical IPs of the BDN is malicious.
  - Any of the client IPs of the BDN is compromised.
  - Any threat intelligence related to the BDN exists.
- All BDNs are malicious.
- Q2: Does the system have an ability to detect unknown malicious BDNs?

  Answer: Leopard can successfully detect unknown malicious BDNs.

Insight into $D_{unknown}$

- Phenomenon: 271 BDNs which come from 87.98.175.85 are meaningless and look like randomly generated. The remaining 15 BDNs are readable.

- It seems that cybercriminals may try to combine the domain generation algorithm (DGA) technique with BDNs. Leveraging DGArchive, we confirmed that BDNs from 87.98.175.85 were generated by Necurs.

Table 7. Examples of the malicious BDNs.

BDNs from 87.98.175.85	BDNs from the other IPs
bafometh.bit	goshan.bit
nenhqlbxxiewmflyckqa.bit	thereis.null
gkgyrwtocxrkrixcxou.bit	log.null
jjffpcvbsyayrluwidxo.bit	ali2.null
lcqpwfvim.bit	systemblink.bit

Outline

1. Background

2. Automatic Detection

3. Evaluation

4. **Limitations**

5. Conclusion

Limitations

- Design
  - Rely on feature engineering and expert knowledge.
  - The system is easily passed by if attackers know features.
  - Rely on “clean” data.
  - Only dealing with BDN-based malware.
- Evaluation
  - The dataset is a little biased due to selecting the top 5K domains of Alexa in the training phase.
  - Lacking effective methods to correctly label benign BDNs.

Outline

1. Background
2. Automatic Detection
3. Evaluation
4. Limitations
5. Conclusion

Conclusion

- We attempt to appeal on researchers to notice the new threat.
- We are the first to propose an automatic detection of malicious blockchain domain names and evaluate it with real-world traffic.
- We get an insight into detected BDNs and discover a variant malware which combined DGA and BDN techniques.
- We present two datasets related to the study of BDN-based malware.

中国科学院 信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

Thanks!

huangzhangrong@iie.ac.cn

Data available at: https://drive.google.com/open?id=1YzVB7cZiMspnTAERBATyvqWKGj0CqGT-

PAM 2020